AD-A113 867 NAVAL OCEAN SYSTEMS CENTER SAN DIEGO CA F/G 9/4
MAXIMUM LIKELIHOOD SEQUENCE ESTIMATION FOR UNKNOWN, DISPERSIVE,--ETC(U)
SEP 81 L E HOFF, R L MERK, S NORVELL
UNCLASSIFIED NOSC-TR-727 NL

1 OF 1
AD A
113867

NOSC

END
DATE
FILMED
5-82
DTIC

(12)

NOSC

NOSC TR 727

NOSC TR 727

AD A113867

Technical Report 727

MAXIMUM LIKELIHOOD SEQUENCE ESTIMATION FOR UNKNOWN, DISPERSIVE, AND TIME VARIANT COMMUNICATION CHANNELS

DTIC
ELECTE
APR 26 1982
S D
E

LE Hoff
RL Merk
S Norvell

30 September 1981

Prepared for
NOSC Independent Exploratory Development Program

Approved for public release; distribution unlimited

DTIC FILE COPY

NAVAL OCEAN SYSTEMS CENTER
SAN DIEGO, CALIFORNIA 92152

04 ... 024

NAVAL OCEAN SYSTEMS CENTER, SAN DIEGO, CA 92152

AN ACTIVITY OF THE NAVAL MATERIAL COMMAND

SL GUILLE, CAPT, USN
Commander

HL BLOOD
Technical Director

ADMINISTRATIVE INFORMATION

The work discussed in this report was done by the RF and Acoustic Communications Technology Branch (Code 8112) under the sponsorship of the Independent Exploratory Development Program ZD84811A01, program element 62766N, subproject ZF66212001.

Released by
MS Kvigne, Head
Communications Research and
Technology Division

Under authority of
HD Smith, Head
Communications Systems and
Technology Department

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER NOSC Technical Report 727 (TR 727)	2. GOVT ACCESSION NO. AD-A113 467	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) MAXIMUM LIKELIHOOD SEQUENCE ESTIMATION FOR UNKNOWN, DISPERSIVE, AND TIME VARIANT COMMUNICATION CHANNELS		5. TYPE OF REPORT & PERIOD COVERED 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) LE Hoff, RL Merk, and S Norvell		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Ocean Systems Center San Diego, CA 92152		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 62766N, ZF66212001
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Material Command Washington, DC 20360		12. REPORT DATE 30 September 1981 13. NUMBER OF PAGES 48
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Viterbi decoder	Hf communications
Kalman filter	Underwater acoustic communications
Widrow-Hoff	Intersymbol interference
LMS algorithms	Channel estimation

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The Maximum Likelihood Sequence Estimation (MLSE) algorithm is one of a number of techniques developed for processing signals with intersymbol interference. The MLSE algorithm is optimum in the sense that it minimizes the probability of error. ~~In~~ this report ~~we~~ describes the intuitive and mathematical basis for the MLSE algorithm. The derivation assumes the channel is known and time invariant.

In Navy communications (underwater acoustic and hf radio channel) channel characteristics are unknown to the receiver. ~~The approach~~ to this problem taken ~~here~~ is to attach to the transmitted message a preamble consisting of a known bit sequence. The receiver uses it to estimate channel response. The MLSE algorithm is applied to the received data using the estimated channel response in lieu of the actual. Performance is then

20. ABSTRACT (Continued)

compared under ideal (known channel characteristics) and realistic (estimated channel characteristics) conditions, as a test on the effectiveness of the channel estimation procedure.

This approach can be extended to channels exhibiting time variations. Updates to the matched filter and metrics are performed so that the channel distortion is more precisely represented by the estimated encoding. This is done periodically by injecting a short, known sequence into the message stream.

The adaptive MLSE algorithm was tested on signals recorded during an hf field test. The results are given in this report.

Accession For	
NTIS GRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A	

DTIC COPY INSPECTED 2

SUMMARY

The underwater acoustic and hf radio channels have dispersive characteristics which inhibit the use of high data rate modems. The refracting and reflecting portions of the transmission media not only distort the shape of the transmitted bits but also add delayed versions. On high data rate communication links these echoes cause the pulses to interfere with each other, resulting in high error rates. The problem is further complicated because the channel characteristics are time as well as geometry dependent.

In order to alleviate these problems communication design engineers of the 1950s limited the channel data rate (bandwidth) so that the pulse duration would be longer than the maximum time delay spread expected; increases in data rate beyond this limit were provided by adding more parallel data channels. Thus, parallel bit streams were an early and practical solution although one which could not overcome frequency selective fading.

More recently pressure has been mounting to return to serial type modems with the advent of EW requirements for wideband (spread spectrum) modulations. As a consequence, techniques for processing these signals have been developed with promising results. One such technique is the Maximum Likelihood Sequence Estimation (MLSE) algorithm, which is optimum for minimizing the probability of error on channels with intersymbol interference. This algorithm was derived assuming ideal conditions; ie, the channel characteristics were assumed known (to the decoder) and time invariant.

We begin this report by describing the intuitive and mathematical basis for the MLSE algorithm. In the context of dispersive channel characteristics, we assume that the dispersive channel can be modeled as a convolutional encoder and with the use of a clever decoder (the Viterbi algorithm) the message bits can be extracted. Under conditions of stationary time statistics and known channel characteristics we show that the MLSE algorithm performs exceptionally well. Under some conditions it is 20 dB better than a 16 parallel tone signaling scheme. In fact, for some nonfading multipath channels, it performs nearly as well as binary PSK on a nondispersive (no multipath) channel.

In the nonideal world of Navy communications (underwater acoustic and hf radio channel) the channel characteristics are unknown to the receiver. One approach to this problem, and the one taken here, is to attach to the transmitted message a preamble consisting of a known bit sequence. The receiver uses this bit sequence to estimate the channel response. The MLSE algorithm is then applied to the received data using the estimated channel response in lieu of the actual. To test the effectiveness of our channel estimation procedures, we compare the performance, in terms of bit errors, of the MLSE under ideal (known channel characteristics) and realistic (estimated channel characteristics) conditions.

We examine two algorithms for estimating the channel response, the Widrow-Hoff LMS and the Kalman Filter (KF) algorithm. Equations describing these algorithms are presented. Each filter has its own advantage. The LMS requires less computational complexity; the KF is more accurate (optimal mean square error filter) and converges more rapidly. An evaluation of each filter is made by simulating the case in which the channel exhibits one delayed bit for every bit sent. The bit error performance, as exhibited by the MLSE algorithm, is essentially equivalent to that achieved previously under the ideal conditions for either the LMS or the KF estimator. Thus, for the specific cases undertaken in this study the use of either LMS or KF has been shown to be practical for estimating channel dispersive characteristics (delayed signal replicas).

The approach of using a preamble for estimating channel response on a time invariant channel can be extended to channels exhibiting time variations. Updates to the matched filter and metrics are performed so that the channel distortion is more precisely represented by the estimated encoding. This is done periodically by injecting a short, known sequence into the message stream.

The adaptive MLSE was tested on signals recorded during an hf field test. These signals exhibit multipath time dispersion and time variations. The results of several of these tests are given in this report to illustrate the effect of time variations on the MLSE performance. Two types of fading are present in the examples: nonfading and fast, shallow fading. The adaptive MLSE can perform nearly as well on nonfading channels as it can on nondispersive, nonfading channels. For the nonfading channel only infrequent channel estimation updates are used. For fast, shallow fading channels, the adaptive MLSE does nearly as well but requires frequent updates to the channel estimate.

For channels with moderate time dispersion and slow fade rates, the adaptive MLSE is an effective technique for providing reliable communication at high data rates. For these channels, the output bit error rate will be lower than either the Decision Feedback Equalizer or the parallel tone modulation scheme. The drawback of the adaptive MLSE is its implementation complexity which grows exponentially with multiplath time dispersion.

Because of the large time dispersion and fast fading rates found on some military communication channels, the adaptive MLSE described here will not be practical. It is recommended that a sequential decoding algorithm with a continuous channel tracking capability be developed. A careful comparison with other signaling schemes for multipath channels, such as Decision Feedback Equalizer and parallel tone modulation formats, would provide valuable information to system design engineers.

CONTENTS

I. INTRODUCTION

Essential to the successful execution of future command and control of the Navy task force is rapid and reliable communication among ships, submarines, and shore elements. Communication nets will be required to carry large volumes of message traffic, data intelligence, and digital secure voice. Communication links, therefore, must be able to operate reliably at high data rates.

However, many of the communication channels that can provide the task force with the communications range and area coverage, such as hf and underwater acoustics, are degraded by multipath and time dispersion. Short bit durations associated with high data rates are typically several times less than the duration of the dispersion. Serial transmission of the bits would result in severe intersymbol interference. Use of demodulators designed for a nondispersive channel would result in an unacceptably high error rate.

Historically, systems designed for these channels use long symbol durations to minimize the pulse distortion and intersymbol interference. The long pulse durations restrict communications to low data rates. To obtain high data rates, several pulses are transmitted in parallel on separate tones. Even though this signal format minimizes intersymbol interference, multipath channels still cause high error rates because of selective frequency fading and doppler spread.

During the past 10 years considerable progress has been achieved toward mitigating intersymbol interference of digital serial transmissions caused by multipath and time dispersion. One of the most impressive algorithms for processing the received signal is the Maximum Likelihood Sequence Estimation (MLSE) algorithm (ref 1). This algorithm is shown to be optimum for intersymbol interference.

In section II we derive the MLSE algorithm by using a waveform encoding point of view. It is felt that this derivation is more intuitive than the previous purely mathematical derivations (ref 1, 2, 3). For this derivation, the channel response is assumed known. Computer simulation results for simple two-path channels are given in section III where the results are compared with the performance of the parallel tone modem now used by the Navy. In section IV we address the realistic situation in which the channel response is unknown. The approach taken is to precede the message with a preamble consisting of a known bit sequence

[1] G David Fornery, Jr, Maximum-Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference, IEEE Transactions on Information Theory, vol IT-18, no 3. May 1972

[2] G Ungerboeck, Adaptive Maximum Likelihood Receiver for Carrier Modulated Datatransmission Systems, IEEE Transactions on Communications, vol COMM-22, no 5, May 1974

[3] G Ungerboeck, Linear Receiver and Maximum-Likelihood Sequence Receiver for Synchronous Data Signals, IEEE International Communications Conference Proceedings, June 1973

and to use this preamble at the receiver to estimate the channel impulse response (ref 4–7). Two estimation algorithms are examined and simulation results are presented. Section V discusses an approach to extend the channel estimation technique to channels that are time variant. We present the results of applying this technique to signals transmitted over an hf link and recorded during an hf field test. Several examples are given to illustrate the effects of time variant multipath channels and nongaussian noise.

II. MLSE ALGORITHM DERIVATION

In this section we derive the MLSE algorithm by drawing an analogy between time dispersive channels and convolutional encoders. The optimum decoding algorithm (minimizing probability of error) for convolutional codes is known to be the Viterbi algorithm. In order to implement the Viterbi algorithm, a recursive relationship must be developed for computing the log likelihood ratio (metric) for the received message. The recursive relationship must be able to update the metric by using only the part of the received waveform that occurred during the last bit interval. Two such recursive algorithms are derived here. They are equivalent but differ in the complexity of their implementation.

The basic model used to represent the data communication system is shown in figure 1. The data sequence, $\{I_n\}$, modulates a basic transmitter pulse, x(t), at a rate of 1/T.

$$S(t) = \sum_{n=0}^{N-1} I_n \, x(t - nT) \tag{1}$$

where T is the pulse duration and I_n the information symbol. Although several types of modulation can be used with the algorithms derived for intersymbol interference, binary phase shift keying (PSK) has been selected here, where I_n will be either +1 or −1.

For multipath dispersive channels the response to the basic transmitted pulse can be written as

$$z(t) = \sum_{\ell=1}^{L_p} \alpha_\ell(t) \, x(t - \tau_\ell) \tag{2}$$

---

[4]Morley, RE, Jr, DL Snyder, Maximum Likelihood Sequence Estimation for Randomly Dispersive Channels, IEEE Transactions on Communications, vol COM-27, no 6, June 1979

[5]Hoff, LE, RL Merk, Soft Decision Demodulation Using the Viterbi Algorithm, NOSC Technical Note 544, September 1978

[6]Norvell, S, Channel Estimation for the HF Channel, NOSC Technical Note 545, September 1978

[7]Hoff, LE, Norvell, S, Adaptive Maximum Likelihood Sequence Estimation for the HF Channel, 13th Asilomar Conference on Circuits, Systems and Computers, November 1979

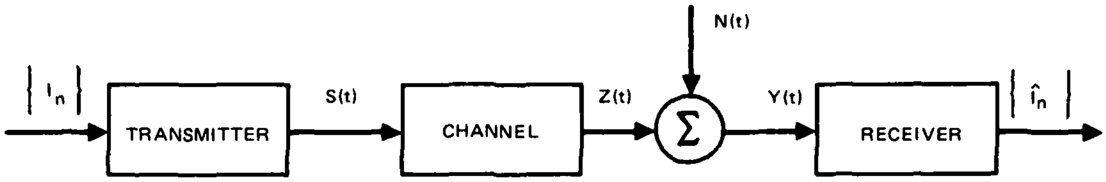


Figure 1. Model of data communication system.

where $\alpha_\ell(t)$ is the complex gain of the ℓ^{th} path at time t, τ_ℓ the time delay of the ℓ^{th} path, and L_p the number of paths. If the channel is time invariant, $\alpha_\ell(t)$ becomes simply α_ℓ, which does not change with time. In this section we derive the MLSE algorithm assuming the channel is time invariant. In section V we extend the algorithm to slowly varying channels and present some results of applying the technique to signals recorded from an hf channel.

A specific message format used for simulating transmitted messages is shown in figure 2. It consists of a noise interval and a signaling interval. The signaling interval consists of a preamble, message text, and a tail sequence. The preamble is known at both the receiver and the transmitter. It is used to set up the matched filter and the metrics and to initialize the Viterbi decoder.

DISPERSIVE CHANNELS AND WAVEFORM ENCODERS

Since we are assuming a time invariant channel the channel response to the basic transmitter pulse x(t) in equation (2) can be written as

$$z(t) = \sum_{\ell=1}^{L_p} \alpha_\ell x(t - \tau_\ell) \tag{3}$$

which is assumed causal, finite in duration, and spanning L + 1 symbol intervals. Equation (3) differs from equation (2) in that α_ℓ is now assumed constant over the interval of interest. Using the principle of linearity, the channel output waveform for the first K + 1 bits is

$$Z(t; I) = \sum_{j=0}^{K} I_j\, z(t - jT) \qquad 0 \leqslant t \leqslant (K + L + 1)T \tag{4}$$

where $0 \leqslant K \leqslant N - 1$. The arguments of z(t; I) show explicit dependence on both time and the transmitted digits, where $I = (I_0, I_1, I_2, \ldots, I_K)$.

Figure 3a is an example of a dispersive channel output. It is very unlikely that such a smooth waveform would occur for the multipath channel, especially for small L_p. This example is not intended to be precise, but easy to visualize. It will be used to illustrate the principles of the MLSE algorithm. Figure 3b shows the real complex envelope of a transmitted waveform for the first five bits, and figure 3c shows a hand-sketched drawing of the dispersed waveform caused by the channel. Additive white noise is not included, only the desired signal. Note that the waveform in figure 3c consists of overlapping shifted versions of the channel response of figure 3b. Each shifted waveform is multiplied by the information bit I_j, which corresponds to that time shift.

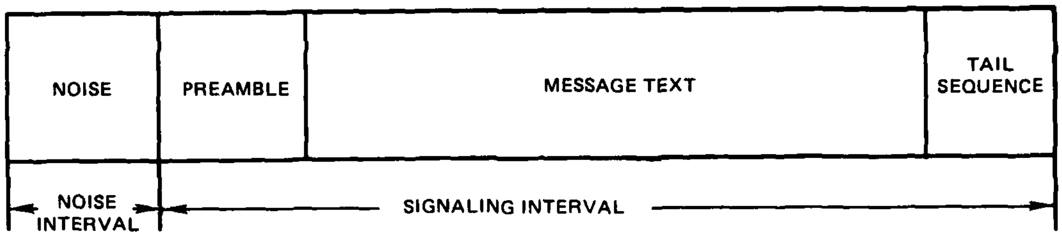
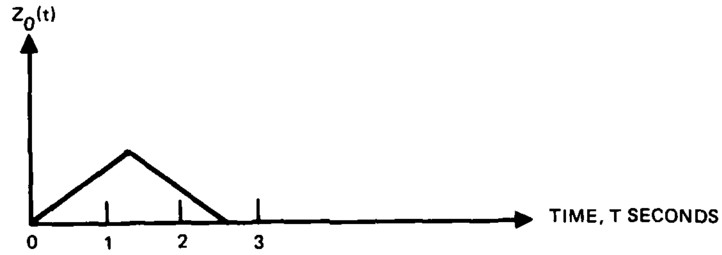
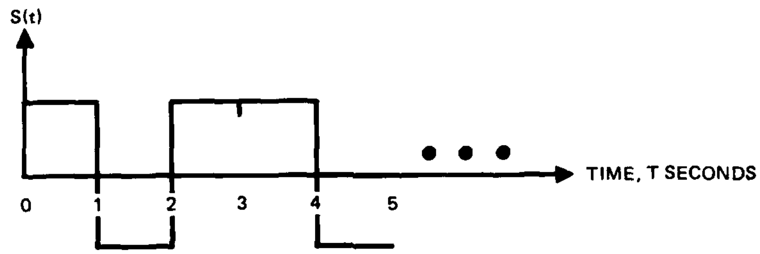


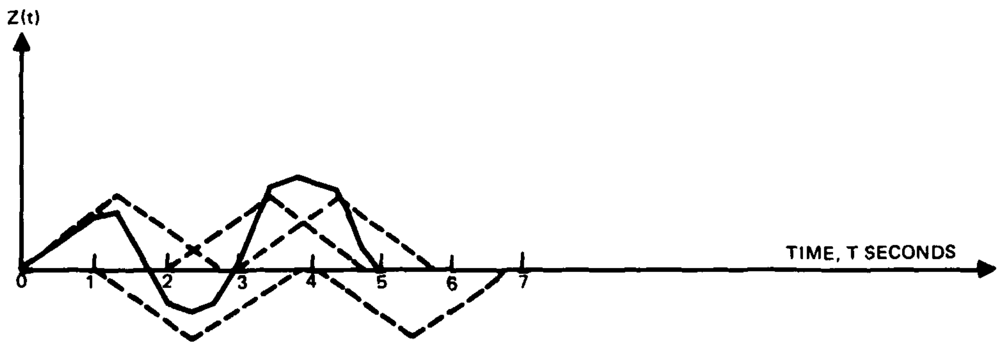
Figure 2. Message format.



a. Channel response to transmitted pulse



b. Transmitted signal waveform (message)



c. Received signal waveform (and components)

Figure 3. Example of intersymbol interference resulting from dispersive channel characteristics.

Figure 4a is another way to draw the dispersed signal waveform. Here the channel response is shown to be shifted and overlapping. A time slice of the dispersive waveform like a dissection of a single interval is composed of a sum of modulated chips. The chips are bit interval partitions of the channel response of figure 3a. In figure 4b the chips are shifted to the same interval and modulated by the sequence of transmitted bits. For example, the chips in the j to j+1 interval consist of chips from bits I_j, I_{j-1}, I_{j-2}. In general, if the channel response is dispersed over the L+1 bits, the chips in the j to j+1 interval will consist of I_j, $I_{j-1}, \ldots, I_{j-L}$. For binary information digits, there are 2^{L+1} waveforms that could be found in the j to j+1 interval, and for an A-ary alphabet there would be A^{L+1}. The collection of waveforms can be treated as an M-ary symbol set where $M=A^{L+1}$.

Let us identify the M waveforms as Q_m (t–jT), m=1, 2, . . . , A^{L+1} or as $Q(t-jT; \underset{\sim}{I}_j)$ to show the explicit dependence on the transmitted sequence, where $\underset{\sim}{I}_j = (I_j, I_{j-1}, \ldots, I_{j-L})$. Outside the interval (jT, (j+1) T), the M-ary waveforms are defined to be zero, and inside the interval some of the waveforms may be the same. The sequence of M-ary waveforms out of the channel up to time (K+1) T depends upon information digits $I_0, I_1, \ldots, I_K$ and can be written

$$Z(t; I) = \sum_{j=0}^{K} Q(t-jT; \underset{\sim}{I}_j) \quad 0 \leqslant t \leqslant (K+1)\,T \tag{5}$$

Figure 5 shows an equivalent model of the transmitter and channel that consists of a shift register and a waveform table look-up. The M-ary waveform in the interval (jT, (j + 1)T) is completely specified by the digits in the register $\underset{\sim}{I}_j = (I_j, I_{j-1}, \ldots, I_{j-L})$. Each time a new digit is transmitted the digits shift to the right and form a new "state." This state specifies the next waveform.

The receiver's job is to observe the received waveform, Y(t), and estimate the transmitted sequence $I_0, I_1, \ldots, I_N$. An equivalent receiver could estimate the states of the shift register by detecting the M-ary waveform. The state of the register is given by the first L digits. The last digit in the shift register, I_{j-L}, can be omitted since it does not contribute to the transition from that state to the next. That is, when a new digit comes into the register, the last digit is shifted out and does not affect the transmitted M-ary waveform.

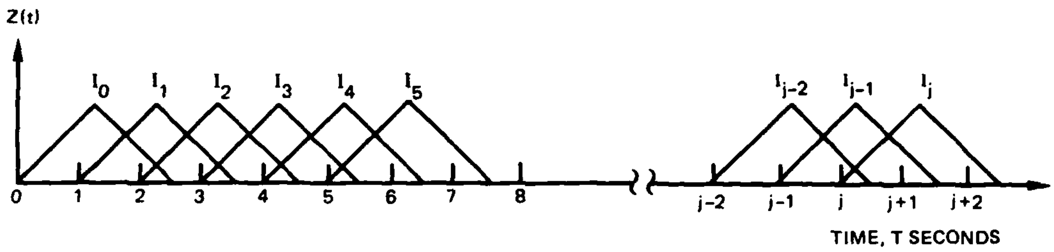


Figure 4a. Hybrid representation for dispersed signal waveform.

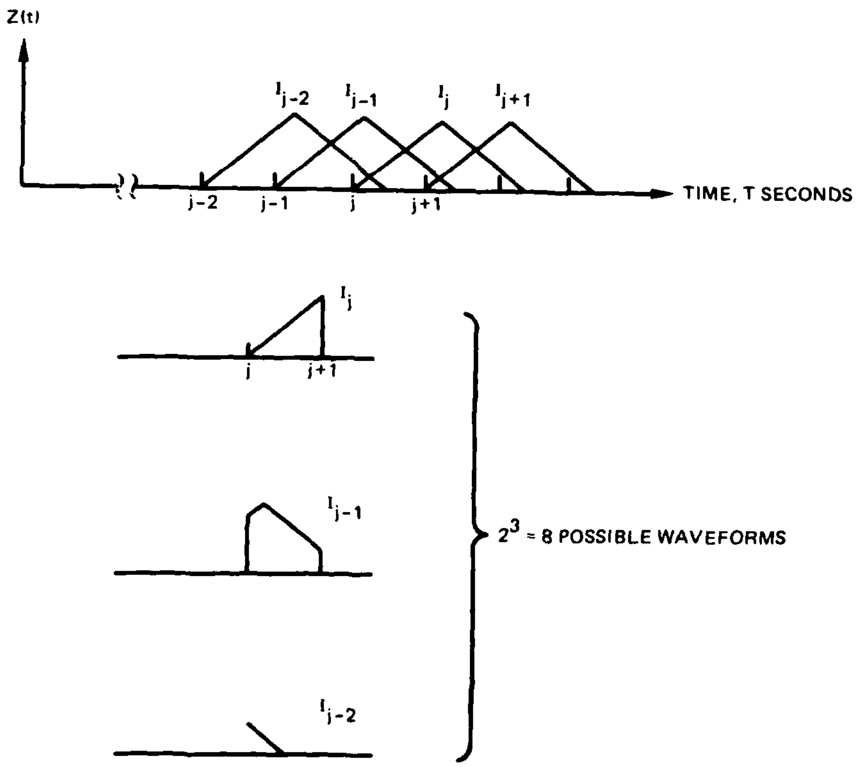


Figure 4b. Chip waveform representation of a time slice of dispersed signal waveform.

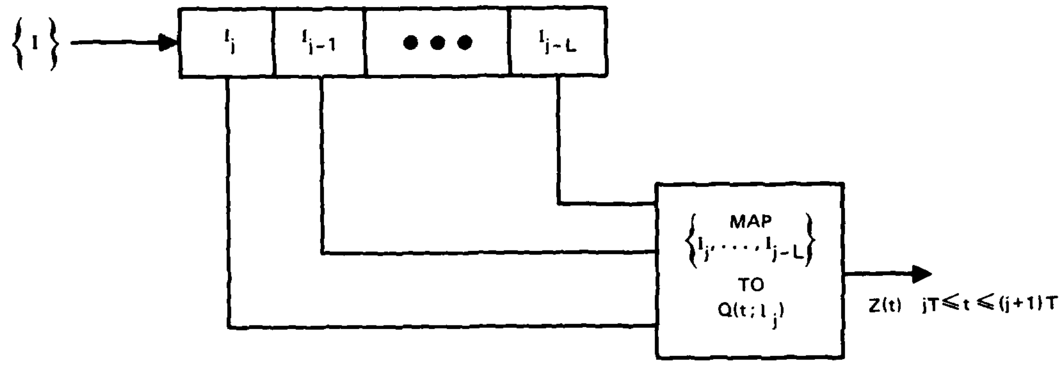


Figure 5. Equivalent transmitter and channel model.

An example of the state transition diagram is given in figure 6 for the example of $L = 2$ shown in figure 3b. The two digits in the circles (nodes) are the digits I_j, I_{j-1} in the register shown in figure 5. When the shift register transitions from one state to the next, a waveform $Q_m(t)$ is transmitted that lasts for one symbol period T. The shift register is clocked every T seconds, so the state diagram changes state every T seconds. As an example, if the register is in state (1,1) and a "-1" enters the register at the left, the register moves to state (-1,1) and $Q_4(t)$ is transmitted during that interval. If the receiver can trace the states of the transmitter shift register, it can decode the message. For example, if the receiver knew the transmitter was in state (1,1), and then received $Q_4(t)$, it would know that the input digit was a -1 and that the new state of the register is (-1,1).

The optimum procedure for decoding the received sequence is the Viterbi algorithm (ref 8). The Viterbi algorithm can be explained in terms of the trellis diagram in figure 7, which is derived by placing the states in figure 6 into a vertical column and repeating to the right for each input. The equivalent transmit/channel encoder starts at the left and moves to the right, one column for each input digit. Let us assume the transmitter precedes the message with L ones, so that when the message starts, the register is in the all-one's state.

The Viterbi algorithm moves to the right one column for each received bit. For each node in the new column, the algorithm compares the metric of the two paths from the previous column that terminate in that node. The algorithm retains the larger of the two metrics and its associated path history. The smaller metric and its path are discarded. Shown in figure 8 is an example of two paths that merge at level (5) in the trellis diagram.

RECURSIVE METRIC ALGORITHM

The metric for one of the nodes in the trellis diagram is the log likelihood ratio for the path starting at the beginning of the message and leading up to that node. Since there are many paths that lead up to a particular node, the metric is the log likelihood ratio of the most likely path. The receiver observes Y(t) and looks for a sequence of transition waveforms corresponding to a particular path through the trellis diagram or state diagram of figure 6. If

[8]Andrew J Viterbi, Convolutional Codes and their Performance in Communications Systems, IEEE Transactions on Communications Technology, vol COM-19, no 5, October 1971

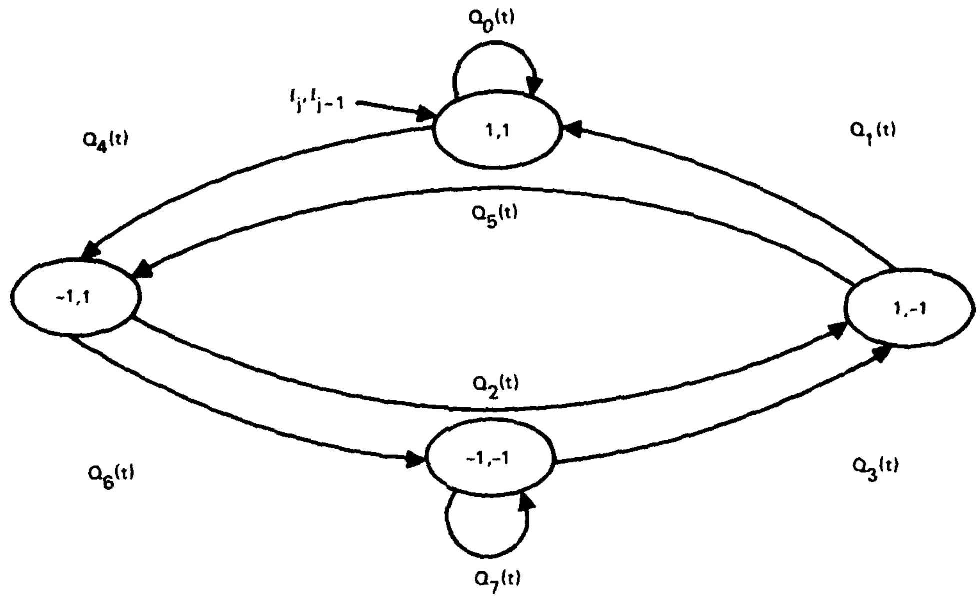


Figure 6. State diagram for L = 2.

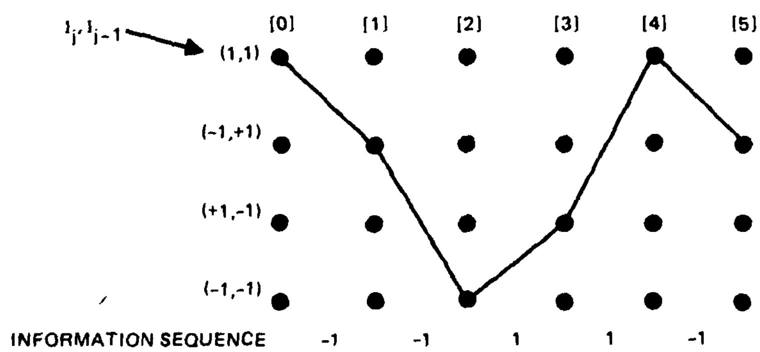


Figure 7. Information path for -1, -1, 1, 1, -1.

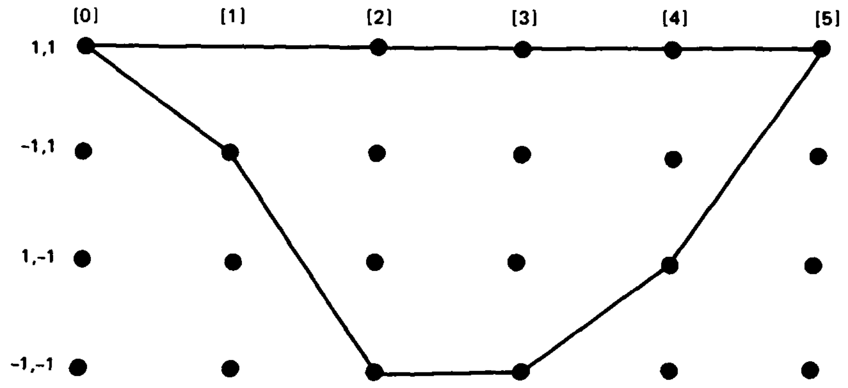


Figure 8. Example of two merging paths.

$Z(t; I)$ in equation (5) is a waveform from $0 < t < (K+1)T$ then the log likelihood ratio (metric) for $Z(t; I)$ is given by Helstrom (ref 9) as

$$M_{K+1}(I_o, I_1, \ldots, I_K) = \text{Re}\,\frac{1}{N}\int_0^{(K+1)T} Z^*(t; I)\; Y(t)\; dt$$

$$-\frac{1}{2}\int_0^{(K+1)T} |Z(t; I)|^2\; dt \tag{6}$$

The problem is to derive a recursive formulation for equation (6) in terms of the transmitted digits so that the metric can be updated at each node as the Viterbi algorithm progresses through the trellis diagram of figure 7.

If we substitute the expression in equation (5) for $Z(t; I)$ we get

$$M_{K+1}(I_o, I_1, \ldots, I_K) - \text{Re}\sum_{j=0}^{K} R_j(I_j) - \sum_{j=0}^{K} C(I_j) \tag{7}$$

where $R_j(I_j)$ is the output of a filter matched to $Q(t-jT; I_j)$ and is given by

$$R_j(I_j) = \int_{jT}^{(j+1)T} Q^*(t-jT; I_j)\; Y(t)\; dt \tag{8}$$

[9]Carl W Helstrom, Statistical Theory of Signal Detection, Pergamon Press. Second Revised edition, Hungary, 1968

Since there are 2^{L+1} different sequences that could be represented by I_j there are 2^{L+1} matched filters.

The variable $C(I_j)$ is the signal-to-noise ratio for the signal waveform $Q(t-jT; I_j)$ and is given by

$$C(I_j) = \frac{1}{2N} \int_0^T |Q(t; I_j)|^2 \, dt \tag{9}$$

Again, there will be 2^{L+1} signal-to-noise ratio parameters.

We can write a simple recursive formula for equation (6) as

$$M_{K+1}(I_o, I_1, \ldots, I_K) = M_K(I_o, I_1, \ldots, I_{K-1}) + \text{Re } R_K(I_K) - C(I_K) \tag{10}$$

The above recursive formula is used to update the metrics in the Viterbi decoding algorithm. It was derived simply by creating a model of the dispersive channel analogous to an M-ary communication system. The difficulty in using this formulation is that the number of matched filters and the number of signal-to-noise ratio parameters grow exponentially with the parameter L+1. This parameter is the ratio of the dispersion length to the length of the bit period. Generation of the matched filters and signal-to-noise parameters for typical dispersions and data rates occurring for the hf and underwater acoustic communication channels could become a computational burden.

The following derivation yields another recursive formula for updating the metrics. This recursive formula requires only one matched filter and L+1 parameters. This is a significant simplification from the M-ary model derivations. However, the derivation is based, not upon a simple intuitive model of the problem, but upon the mathematical relationships of the multipath signals (equations 3 and 4) and the likelihood ratio (equation 6). In the above derivation we used equation (5) to define $Z(t; I)$.

However, $Z(t; I)$ is also given by equation (4) up to time $(K + 1)T$. Equation (5) is zero beyond $(K + 1)T$, whereas equation (4) contains a "tail" for $t > (K + 1)T$. However, our integrals stop at $(K + 1)T$ and the tail will not be a problem. Substituting equation (4) into equation (7) gives

$$M_{K+1}(I_0, I_1, \ldots, I_k) = \text{Re} \sum_{j=0}^{K} I_j^* r_j - \sum_{i=0}^{K} \sum_{j=0}^{K} I_i^* I_j C_{i-j} \tag{11}$$

where we have defined r_j and C_{i-j} as

$$r_j = \begin{cases} \dfrac{1}{N} \displaystyle\int_{jT}^{(j+L+1)T} z^*(t - jT)\, Y(t)\, dt & \text{for} \leq K - L \\[2ex] \dfrac{1}{N} \displaystyle\int_{jT}^{KT} z^*(t - jT)\, Y(t)\, dt & \text{for } K - L \leq j \leq K \end{cases} \tag{12}$$

and

$$C_{i-j} \triangleq \frac{1}{2N} \int_0^{(K+1)T} z^*(t - iT)\ z(t - jT)\ dt \tag{13}$$

Note that $C_{i-j} = 0$ when $|i - j| > L$ and C_{i-j} is Hermitian $C_{i-j} = C^*_{j-i}$

The metric in equation (11) can be written as

$$M_{K+1} (I_o, I_1, \ldots, I_K) = \text{Re} \sum_{j=0}^{K-1} I_j^* r_j - \sum_{i=0}^{K-1} \sum_{j=0}^{K-1} I_i^* I_j C_{i-j}$$

$$+ \text{Re}\ I_K^*\ r_K - 2\ \text{Re} \sum_{j=0}^{K-1} I_K^* I_j C_{K-j} - |I_K|^2\ C_o) \tag{14}$$

The previous metric is given by

$$M_K(I_o, I_1, \ldots, I_{K-1}) = \text{Re} \sum_{j=0}^{K-1} I_j^* r_j - \sum_{j=0}^{K-1} \sum_{j=0}^{K-1} I_i^* I_j C_{i-j} \tag{15}$$

Substituting equation (15) into equation (14) gives the recursive relationship we seek:

$$M_{K+1}(I_o, I_1, \ldots, I_K) = M_K(I_o, I_1, \ldots, I_K)$$

$$+ \text{Re}\ I_K^*\ (r_K - 2 \sum_{j=1}^{L} I_{K-j}^* C_j - I_K\ C_o) \tag{16}$$

where we have simplified the recursive increment term by taking into account the fact that $C_j = 0$ for $|j| > L$.

The recursive relationship in equation (16) requires only one matched filter and L+1 parameters. At time (j+1)T the receiver samples the output of a filter matched to the channel

response and calls this sample r_j. Equation (16) is then used to compute the two (for the binary case) paths into the next node. The path history with the largest metric is retained and updated by the value of I_j.

In this section, we have shown that the communications channel can be modeled as a convolutional encoder whose outputs are analog M-ary waveforms instead of sequences of binary digits. It is known that the maximum likelihood decoder for convolutional codes is the Viterbi decoder. However, to implement the Viterbi decoder algorithm, a recursive relationship for the log likelihood ratio must be known. We have derived a recursive algorithm in equation (16) for the log likelihood ratio for multipath dispersive channels. By implementing it via Viterbi decoding, we have the optimum signal processing algorithm for channels with intersymbol interference, the Maximum Likelihood Sequence Estimate (MLSE) algorithm, as Fornery (1) called it.

The complexity of an algorithm is a measure of how fast a processor must run, or alternatively, how much circuitry is required to implement it. For the MLSE the complexity grows exponentially with the spread factor L, which is the product of the channel time spread, T, in seconds and the information bit rate, R, in bits per second.

The next step is to examine the performance of the MLSE algorithm under the assumptions under which it was developed. The assumptions include perfect knowledge of the channel response, time invariance and an upper limit on the channel dispersion. This will be accomplished by simulating the channel itself and the MLSE algorithm on a computer.

III. MLSE PERFORMANCE FOR KNOWN CHANNEL CHARACTERISTICS

In the following analysis we examine the performance of the MLSE over simulated hf channels. The simulated channels will be simple two-path channels. The performance characteristics of the MLSE algorithm are compared with the parallel tone DQPSK (ref 10) over a multipath channel and these are then compared to optimum performance over a channel which exhibits no intersymbol interference. To simplify the analysis we have made several assumptions about our simulated hf channel: the channel is time invariant (ie, no fading), the channel impulse response is known, and the noise is strictly additive white Gaussian noise.

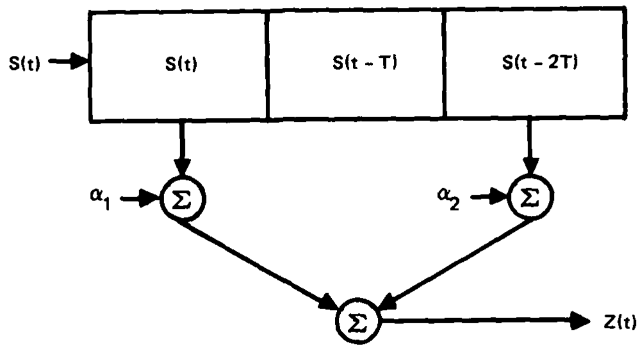
A Monte Carlo simulation is employed to simulate both the pseudorandom binary signal and the pseudorandom Gaussian noise samples. The simulation was run on the NOSC Univac 1110 computer.

SIMULATED CHANNEL

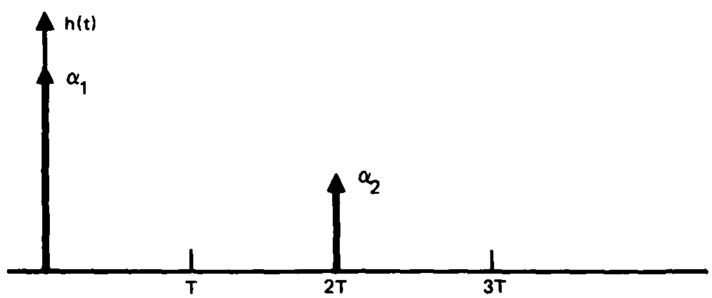
A general two-path channel may be modeled in several different ways, three of which are illustrated in figure 9. In figure 9a we represent the output of the channel by the output of a tapped delay line. The output, z(t), is a linear combination of the input and replicas of the input delayed by 2T seconds. Next, for ease of mathematical manipulation, the impulse response shown in figure 9b is convenient. Here the output can be represented by the convolution of the input with the impulse response. The ray trace diagrams of figure 9c give

---

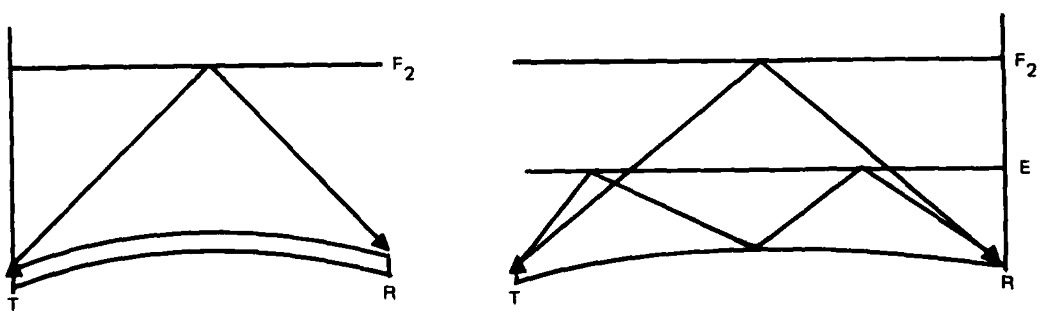
[10] ML Doelz, ET Heald, and DL Martin, Binary Data Transmission Techniques for Linear Systems, Proceedings IRE, vol 45, 656-661, May 1957



a. Tapped delay line



b. Impulse response



c. Ray trace diagrams

Figure 9. Three models illustrating a two-path channel.

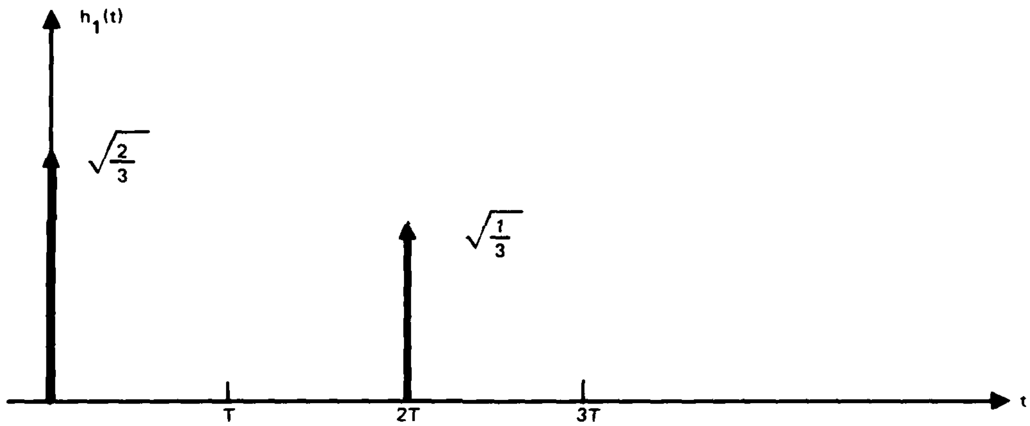
physical insight into how this channel may occur. Figure 10 shows the specific impulse responses used in the simulations.

Figure 11 graphically displays the amplitude spectrum as a function of frequency for the channel shown in figure 10a. Performance of the parallel tone DQPSK on this channel is determined primarily by the two nulls in the amplitude spectrum. Most of the errors occur in the nulls where the signal-to-noise ratio is significantly less than at the other frequencies.

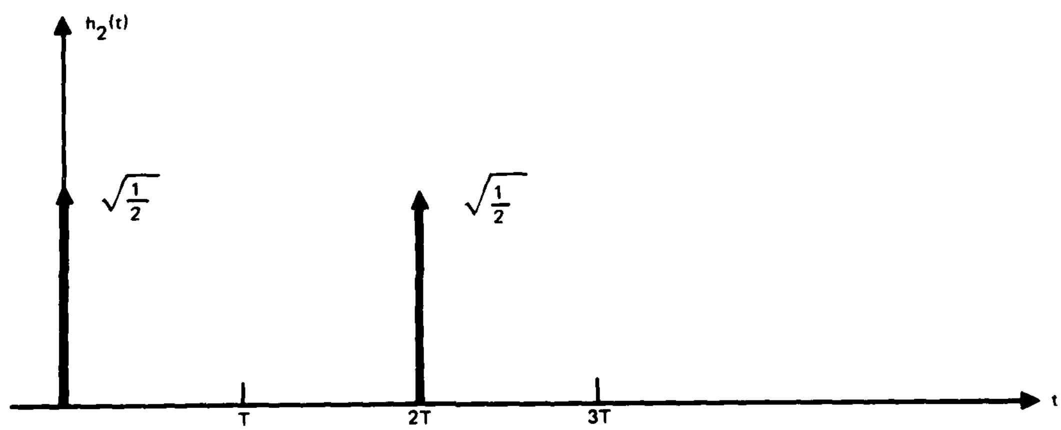
The bit error rate for parallel tone DQPSK is calculated by averaging the bit errors of each tone over all the tones; ie, with 16 tones.

$$E\left[P_{B_e}\left(E_b/N_o\right)\right] = \frac{1}{16}\sum_{j=1}^{16} P_{B_e}\left(E_b/N_o\,,\, j^{th}\text{ tone}\right) \tag{17}$$

Since the probability of a bit is 2/3 the probability of a symbol error, all one needs is the symbol error rate of the j^{th} tone. Unfortunately, no simple exact expression exists for the



a. $\alpha_1 = \sqrt{2/3}$ and $\alpha_2 = \sqrt{1/3}$



b. $\alpha_1 = \alpha_2 = \sqrt{1/2}$

Figure 10. Channel impulse response for two conditions of simulation.

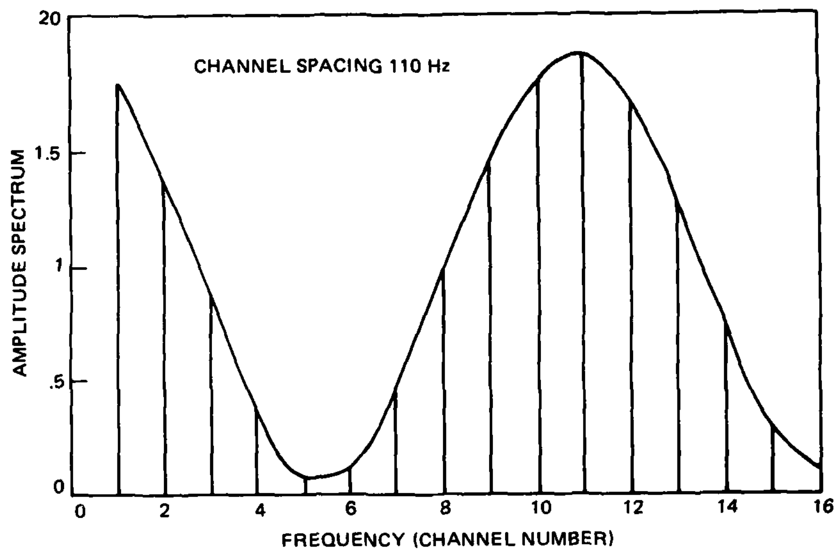


Figure 11. Amplitude-frequency response for a 16-channel, parallel tone, DQPSK modulation for the channel response of figure 10a.

symbol error rate. For large signal-to-noise ratios (larger than 5dB) a good approximation was derived by Cahn (ref 11). By his approximation, the average bit error rate for 16 parallel tone DQPSK becomes

$$E\left[P_{B_e}\left(E_b/N_o\right)\right] = \frac{1}{16}\left(\frac{2}{3}\right)\sum_{j=1}^{16} \operatorname{erfc}\left(\sqrt{\frac{2E_b(j)}{N_o}}\ \sin\frac{\pi}{8}\right) \tag{18}$$

$$\text{where } \operatorname{erfc}(z) = \frac{2}{\sqrt{\pi}}\int_z^{\infty} e^{-t^2}\, dt\ , \tag{19}$$

$$E_b(j) = \frac{E}{N_o}\ (1 + 2\alpha_1\,\alpha_2\ \cos j\,\alpha\tau).$$

$E(\cdot)$ is the expectation operator and

$\alpha = \frac{2\pi}{T}$ is the differential time of arrival.

PERFORMANCE EVALUATION

As a consequence of the above analysis a bit error performance comparison was made between the parallel tone, the serial bit stream (MLSE), and the ideal (no multipath) binary

[11] CR Cahn, Combined Digital Amplitude and Phase Modulation, IRE Transactions on Communications Systems, vol CS-8, p 150-155, September 1960

BPSK approaches. The results of this comparison are shown in figure 12. The impulse response of the channel is shown in the upper right of each graph. The data rate for each case is 2400 bits per second.

The curves illustrate the poor performance of the parallel tone approach. The serial bit stream approach using the MLSE has a 20-dB advantage over the parallel tone design and performs essentially as well as BPSK under ideal (no multipath) conditions.

IV. MLSE PERFORMANCE FOR UNKNOWN CHANNEL CHARACTERISTICS

The MLSE developed up to this point assumes that z(t) is known. This z(t) is used to set up the matched filter and the autocorrelation function C_{jk} used in the log likelihood ratio for the recursive calculation of the metrics. In this section we develop a model for characterizing the channel response, z(t), and algorithms for estimating the parameters used to characterize this z(t).

The output of the channel can be represented as the sum of time shifted versions of the input signal as in equation (2), which is repeated here

$$z(t) = \sum_{\ell=1}^{L_p} \alpha_\ell(t)\, x(t-\tau_\ell) \qquad (2)$$

To put the problem in a form which is easily characterized, several assumptions are made:

(1) Assume that $\alpha_\ell(t)$ can be treated as a constant over the interval of interest; ie, updates of the channel response will be made faster than the rate of change of the channel.

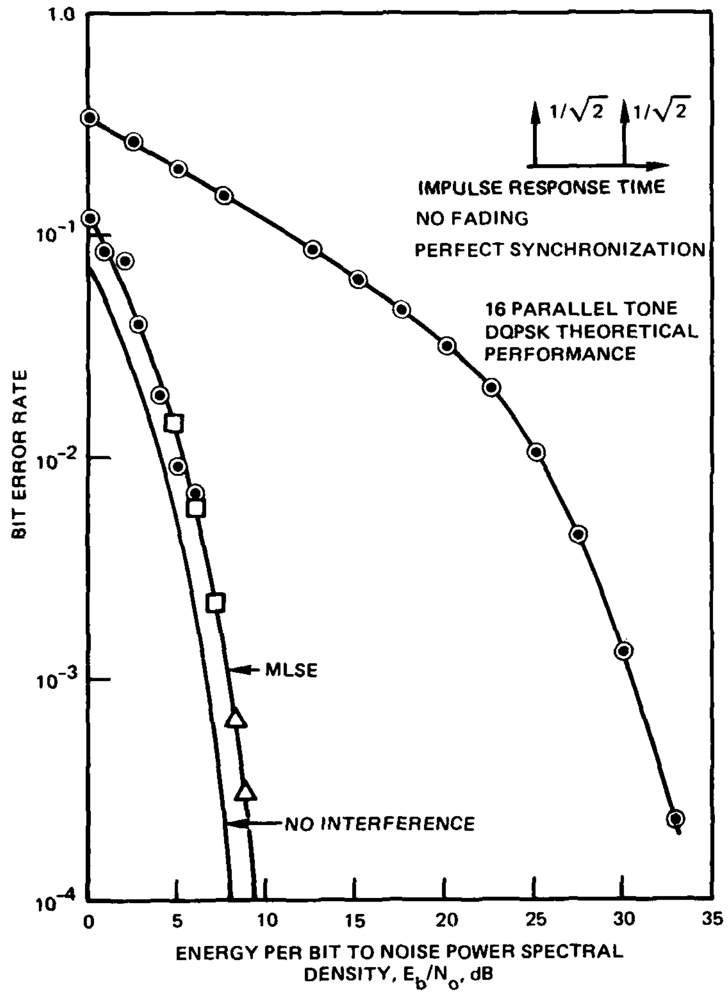
(2) Assume that samples of the input $s(k) = S(t)\,|t = kT$ and output $y(k) = Y(t)|_{t=kT}$ are available for a period known as the preamble.

(3) Assume that the received output can be represented as

$$y(k) = \sum_{\ell=0}^{L} g_\ell(k)\, S(k-\ell) + n(k)$$

$$= z(k) + n(k) \qquad (20)$$

Note that, in general, $g_\ell(k)$ is not equal to $\alpha_\ell(t)$ in equation (2). The $g_\ell(k)$ are the complex tap gains for the uniformly spaced tapped delay line. The $\alpha_\ell(t)$ are the complex gains which correspond to the nonuniformly spaced delays, τ_ℓ. A restriction on L is that it must satisfy the requirement that L times the sampling interval, T, is greater than or equal to the maximum possible signal delay for all signals of interest. The representation of the channel response as the output of a tapped delay line is shown in figure 13. The parameters $g_\ell(k)$ are the tap gains which are complex and may be of magnitude less than one.



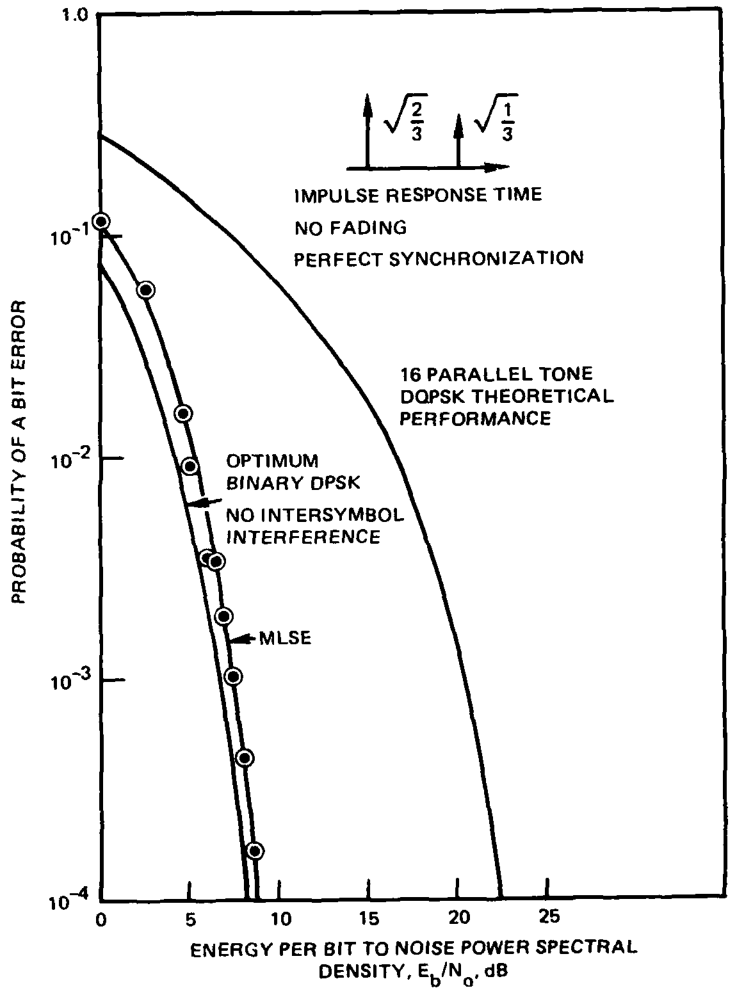


Figure 12. Bit error rate performance of parallel tone, serial data (MLSE), and idealized (nondispersive channel) BPSK for two cases of channel impulse response.

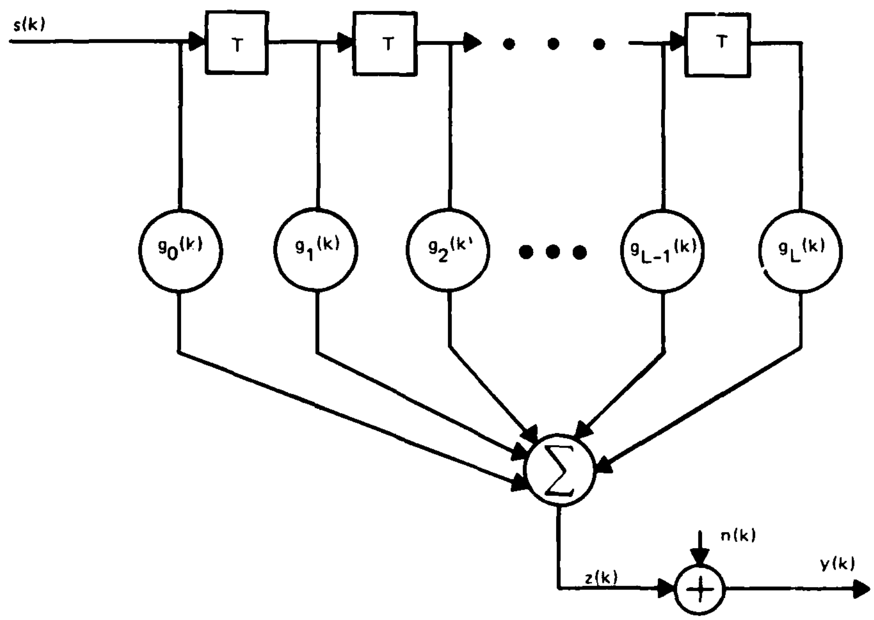


Figure 13. Tapped delay line representation of channel response.

Once we have set up this structure the output response can be estimated by estimating

$$z(k) = \sum_{\ell=0}^{L} g_\ell(k)\, s(k-\ell) \tag{21}$$

So, the problem reduces to estimating the tap gains $g_\ell(k)$, $\ell = 0, 1, \ldots, L$ from the received sequence y(k). Once L and the sampling interval are known, the channel response can be completely specified by g(k). The Kalman filter algorithm (KF) and a least mean squares algorithm (LMS) will be used as channel estimators. The channel estimators characterize the channel by estimating the tap gains $g_\ell(k)$. From these estimated tap gains, estimates of the channel response can be formed. These can be compared to the actual output of the channel. The criterion used in this section to compare the performance of the estimation algorithms is error squared averaged over several simulations of the same channel. As yet there is no analytical correspondence between the mean squared error of the estimation algorithms and the performance of the MLSE. In reference (12) the performance of the MLSE when used with "mismatched" channel estimates is discussed. However, reference 12 addresses only the case in which a specific mismatch occurs. Further work needs to be done for the case of random mismatches which occurs when using an estimation algorithm on time variant channels. The estimates from one of these channel estimators will be used as parameter inputs to a matched

[12] Pennoyer, BL, Performance of an Adaptive or Mismatched Maximum Likelihood Sequence Estimation Receiver, PhD thesis, University of Southern California, February 1977

filter and a maximum likelihood *sequence estimator whose output is the estimate of the* information sequence. This systems is shown in figure 14.

CHANNEL ESTIMATOR ALGORITHMS

In this section the general equations for the KF and the LMS are presented, and these are applied to the channel estimation problem. Since we are assuming that the channel is varying slowly with time we can view the tap gains as constants over our interval of interest.

$$g(k) = g(k - 1) \tag{22}$$

where g(k) denotes the vector of tap gains at time k

$$g^T(k) = [g_0(k)\, g_1(k) \ldots g_L(k)]$$

The measurement y(k) is simply the sampled values of the received signal. Then we have

$$\begin{aligned} y(k) &= z(k) + n(k) \qquad (23) \\ &= H^T(k)\, g(k) + n(k) \end{aligned}$$

where H(k) is the vector of delayed inputs.

$$H^T(k) = [s(k)\, s(k - 1) \ldots s(k - L)]$$

and n(k) is a white, zero-mean random noise sequence with variance R(k).

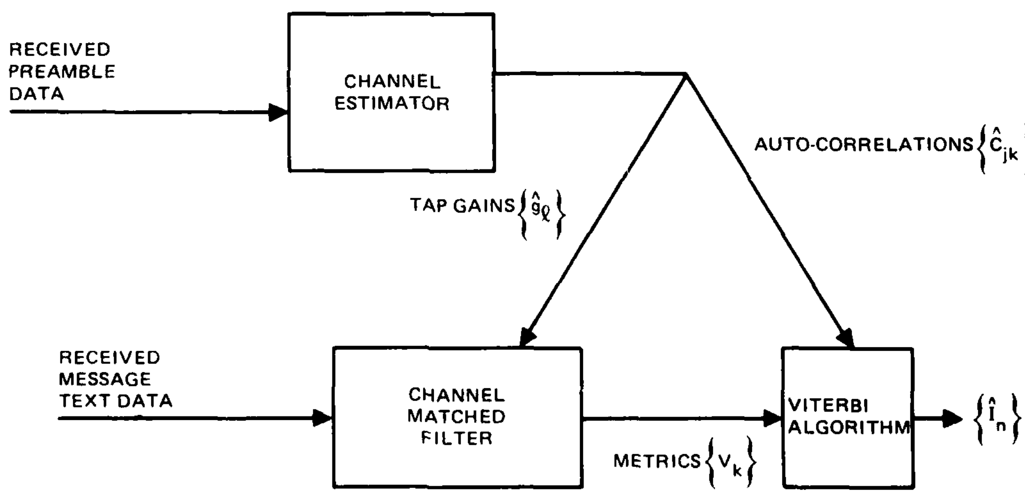


Figure 14. Receiver model.

Kalman Filter Algorithm

Several additional assumptions are needed before the Kalman filter can be implemented. The vector w(k), known as the plant noise, is added to equation (22).

$$g(k) = g(k-1) + w(k-1) \tag{24}$$

This quantity is introduced in order to compensate for numerical precision problems of the system. The noise sequence w(k) is assumed to be zero-mean with covariance matrix Q(k). The initial conditions of the tap gains, the measurement noise and the plant noise are all assumed to be uncorrelated with each other. The basic Kalman filter algorithm used is the following (ref 13, 14):

(1) Initialize filter parameters. Set k = 0.

(2) Increment index k = k + 1

(3) Form projected estimate of covariance matrix

$$V'(k) = V(k-1) + Q(k-1) \tag{25}$$

(4) Compute Kalman gain

$$K(k) = V'(k)\, H(k)\, (H^T(k)\, V'(k)\, H(k) + R(k))^{-1} \tag{26}$$

(5) Form estimates of tap gains

$$g(k) = g(k-1) + K(k)\, [y(k) - H^T(k)\, g(k-1)] \tag{27}$$

(6) Compute covariance of estimate

$$V(k) = [I - K(k)\, H^T(k)]\; V'(k)] \tag{28}$$

Go to step (2)

Initial estimates of all tap gains were taken to be zero. The initial covariances of the estimates were taken to be large to correspond to the case that very little was known about the initial estimates. This ensures that the initial estimates will not be weighted too heavily. The plant covariance matrix Q(k) was taken to be a small positive scalar times the identity matrix to compensate for modeling errors and computational errors accumulated when run on a digital computer with finite word size. The squared error between the estimated output and the actual output averaged over 100 simulations is plotted in figure 15 for the simple two-path channel shown in figure 10a.

[13] HW Sorenson, Kalman Filtering Techniques, Advances in Control Systems, vol 3, p 219-292 1966

[14] HW Sorenson, Filtering and Random Processes in Control, class notes

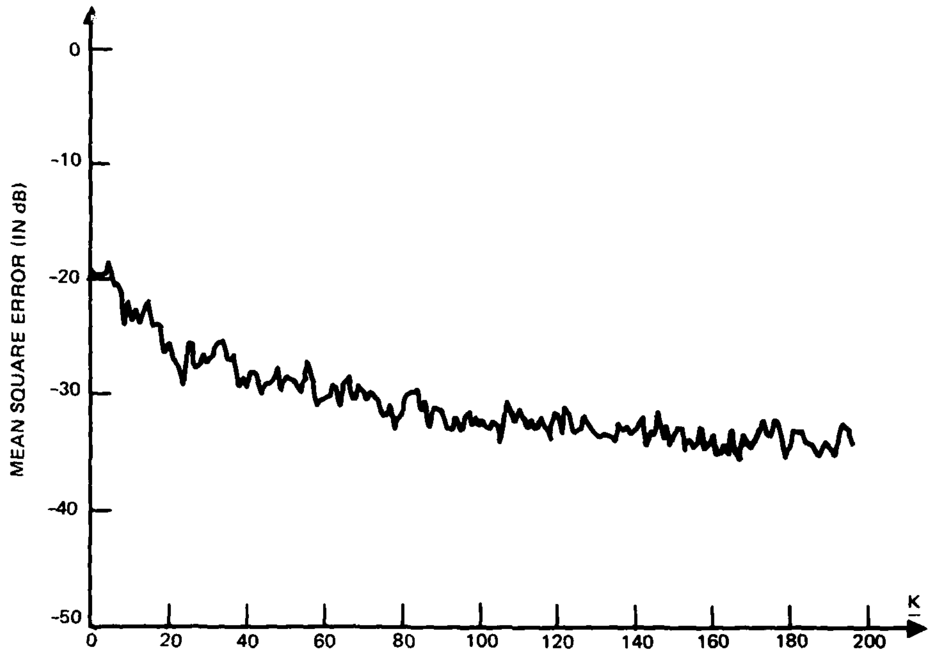


Figure 15. Channel 1, KF.

LMS Algorithm

The basic LMS algorithm used is the following (ref 15, 16):

(1) Initialize parameters

(2) Form error in estimate

$$e(k) = y(k) - H^T(k)\, g(k) \tag{29}$$

(3) Form new estimate

$$g(k+1) = g(k) + 2\,\mu\, e(k)\, H(k) \tag{30}$$

(4) Increment index. Go to (2)

As for the Kalman filter, the initial tap gain estimates were taken to be zero. Unlike the Kalman filter, the LMS filter requires no initial covariance estimate or plant covariance matrix.

---

[15] B Widrow, et al, Stationary and Nonstationary Learning Characteristics of the LMS Adaptive Filter, Proceedings of the IEEE, vol 64, no 8, 8 August 1976, p 1151-1162

[16] B Widrow, Adaptive Filters I: Fundamentals, SEL-66-126 Stanford Electronics Laboratories, Stanford, California, December 1966

For the LMS, the scalar parameter μ is a convergence factor that controls stability and rate of adaptation. A sufficient condition on μ for convergence is

$$0 < \mu < \frac{1}{L+1} \tag{31}$$

Two other criteria used for deciding the value of μ to be used are its effect on filter misadjustment and speed of adaptation. The misadjustment M due to noise is defined as the ratio of average excess mean square error (mse) to minimum mse

$$M = \frac{\text{average excess mse}}{\text{minimum mse}}$$

For the LMS algorithm, under the conditions assumed above

$$M = \mu(L+1) \tag{32}$$

The time constant of the filter, τ, can be written as

$$\tau = \frac{1}{4\mu} \tag{33}$$

Tradeoffs between steady-state mean square error and rate of adaptation must be made. This is done by selecting the appropriate value of μ. Figure 16 demonstrates the effect of two different values of μ on the same channel. The squared error is averaged over 100 realizations. The value of μ = .01667 converges faster than a value of μ = .008335. At the same time a value of μ = .01667 has a higher mean square error than does a value of μ = .008335.

The LMS and KF estimation algorithms are compared in figure 17. There are advantages and disadvantages to each of them. The Kalman filter converges more rapidly to a smaller mean square error than does the LMS filter. However, this improvement in performance has been obtained at the cost of computational complexity. The Kalman filter requires computations, at each iteration, on the order of the square of the number of parameters to be estimated. The LMS only requires a linear multiple of the number of parameters to be estimated but does not converge very rapidly nor does it converge to the optimal estimate. Another problem encountered when using the LMS is in the choice of a value for μ. For each preamble length and desired accuracy (in terms of mean square error) a best value of μ can be found, but this value may produce very poor results when used on another case. The Kalman filter does not suffer from this drawback.

EVALUATION OF MLSE WITH ESTIMATOR ALGORITHMS

This section examines the performance of the Maximum Likelihood Sequence Estimator when estimates of the channel parameters are used instead of the actual channel parameters. The derivation of the MLSE in section II and the performance curves of section III were obtained under the assumption that the channel was known and time invariant. The Viterbi algorithm from section II will be combined with the estimation algorithms from the first part of this section to examine the performance of the MLSE over an unknown (but

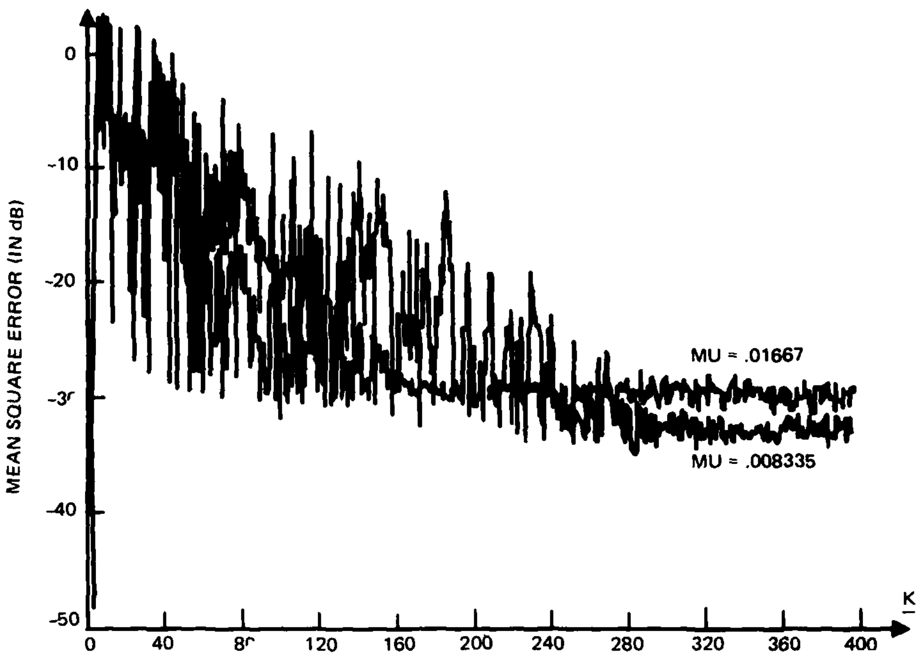


Figure 16. Channel 1 LMS μ = .01667 vs. μ = .008335.

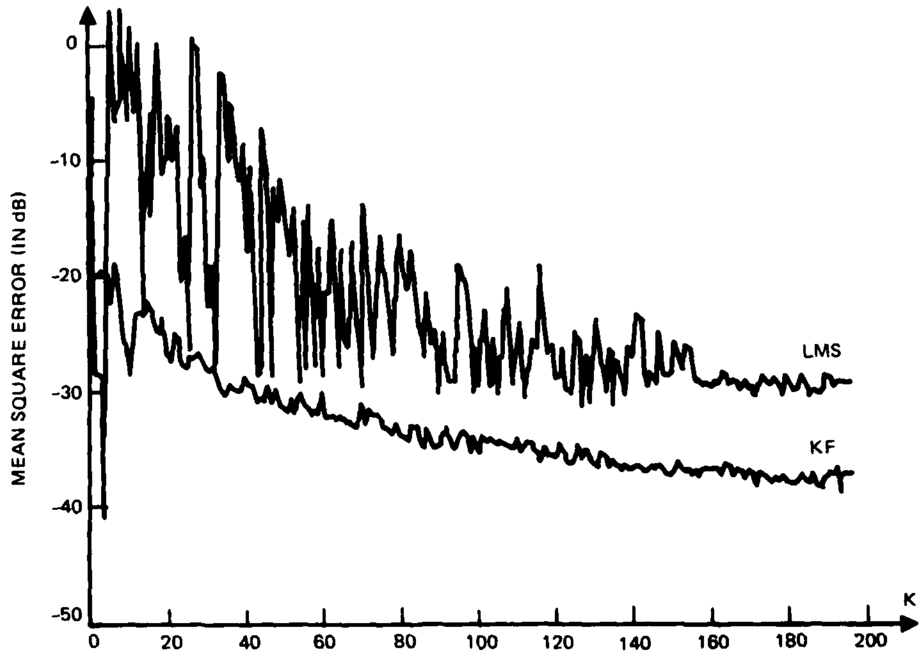


Figure 17. Channel 1 LMS μ = .01667 vs KF.

still time invariant) channel. The message format used, shown in figure 2, includes a preamble. This preamble, which is a PN sequence known at both the transmitter and receiver, is used to estimate the channel response. The channel estimate is used to set up the matched filter and initialize the metrics which the Viterbi decoder requires. The problem is that these channel estimates will never exactly equal the actual channel parameters. They will fluctuate around the actual values. This section will show, for specific cases, how the estimates affect the performance of the MLSE. Upper bounds on the performance of the MLSE with mismatched channel estimates are given in reference (17). In this section actual simulation results are given.

We have taken two approaches to implementing a channel estimator. One, the Kalman filter, achieves a better estimate in shorter time but at the expense of a heavier computational burden. The other, the LMS filter, does not impose so heavy a computational burden but takes longer to converge to a poorer estimate. First, the performance of the MLSE with a long preamble will be examined. Both Kalman filter and LMS filter will be simulated. Next, the performance of the MLSE using the Kalman filter and the LMS filter over a much shorter preamble will be examined. Then the performance of the MLSE with the LMS estimator will be examined in more detail. It will be shown that in addition to the problem of slow convergence to suboptimal estimates the LMS suffers from the problem of choice of suitable values for μ.

Figure 18 shows the performance of the MLSE with a preamble of length 1000 bits. The channel impulse response for the performance curves is shown in the upper right corner for each graph. These simple two-path channels are the same as the simulated channels utilized in previous sections of this report. The performance of the Viterbi decoder with the LMS estimator is compared with that of the Viterbi decoder with the Kalman estimator for preambles of length 1000. This a relatively long preamble. The Kalman filter approaches the optimal estimate as the length of the preamble increases. Thus, the performance of the MLSE with a Kalman filter and long preamble should approach the performance of the MLSE over a known channel. For a preamble of this length using the LMS estimator the appropriate value of μ can be chosen to minimize the mean square error. This can only be done with a long preamble since the rate of convergence of the LMS filter is linearly proportional to the value of μ while the residual mean square error is inversely proportional to μ. Figure 18 shows for this length preamble the LMS does nearly as well as the Kalman filter. The conclusion to be drawn from this is that for a very long preamble the LMS will perform nearly as well as the Kalman as long as the value of μ is chosen wisely. As will be demonstrated later it is important that the appropriate value of μ be chosen when using the LMS filter. Use of a long preamble demonstrates that the problem of the unknown channel characteristics can be solved.

In many situations a long preamble is not a practical solution to the problem of estimating the channel response for the MLSE. A much shorter preamble will often be required. As a result, the effect of a much shorter preamble length will be discussed. Figure 19 shows the performance of the MLSE with a preamble length of 50 using both the LMS and the Kalman filter estimation algorithms. Once again the same two-path channels are used. The performance of the Kalman with preamble length 50 is degraded slightly from the case of the Kalman with preamble length 1000. The difference is no more than 0.5 dB. In many cases the decrease in overhead and computational burden that is achieved by using the

[17] Magee, FR Jr, JG Proakis, Adaptive Maximum-Likelihood Sequence Estimation for Digital Signaling in the Presence of Intersymbol Interference, IEEE Transactions on Information Theory, vol IT-19, January 1973

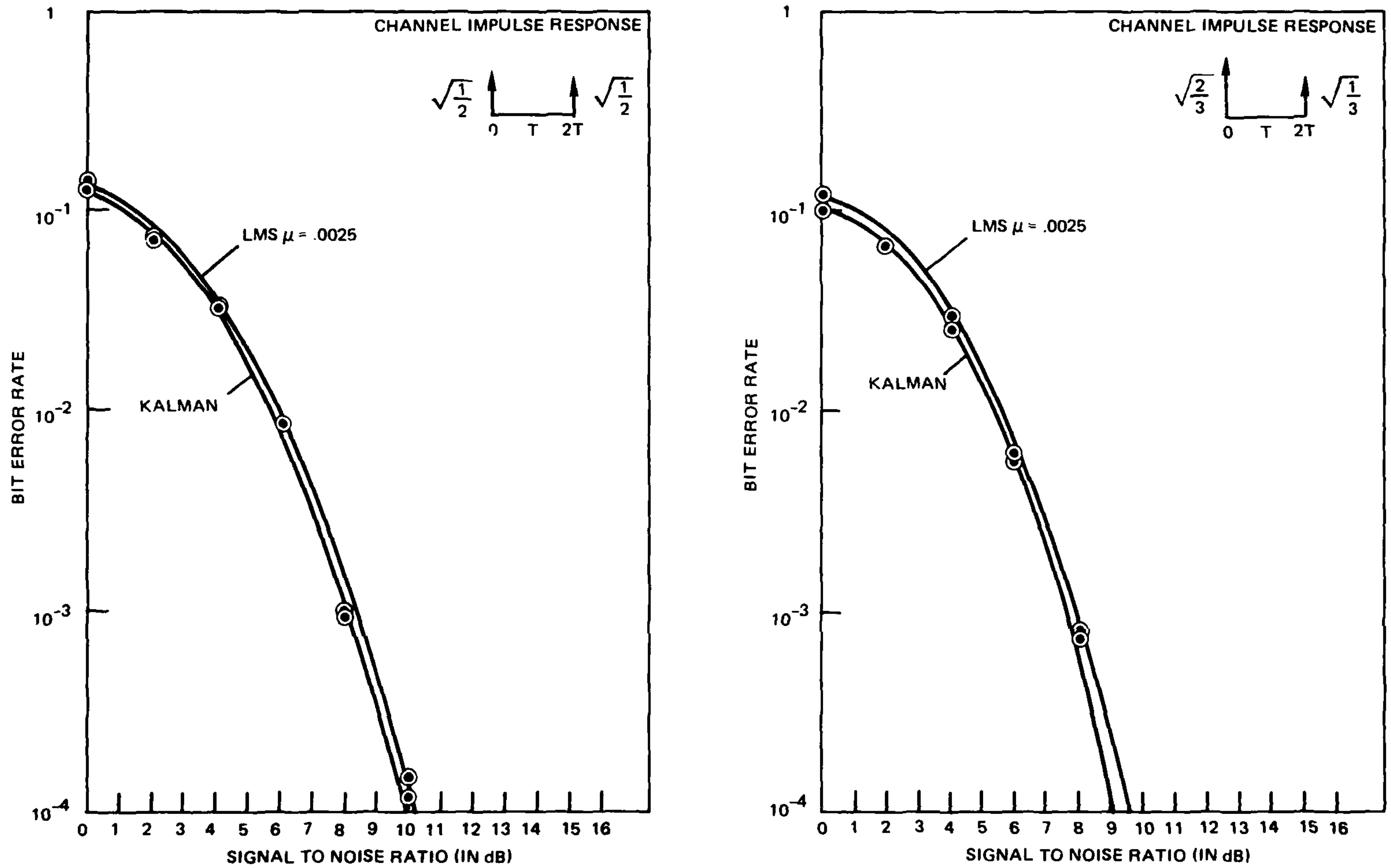
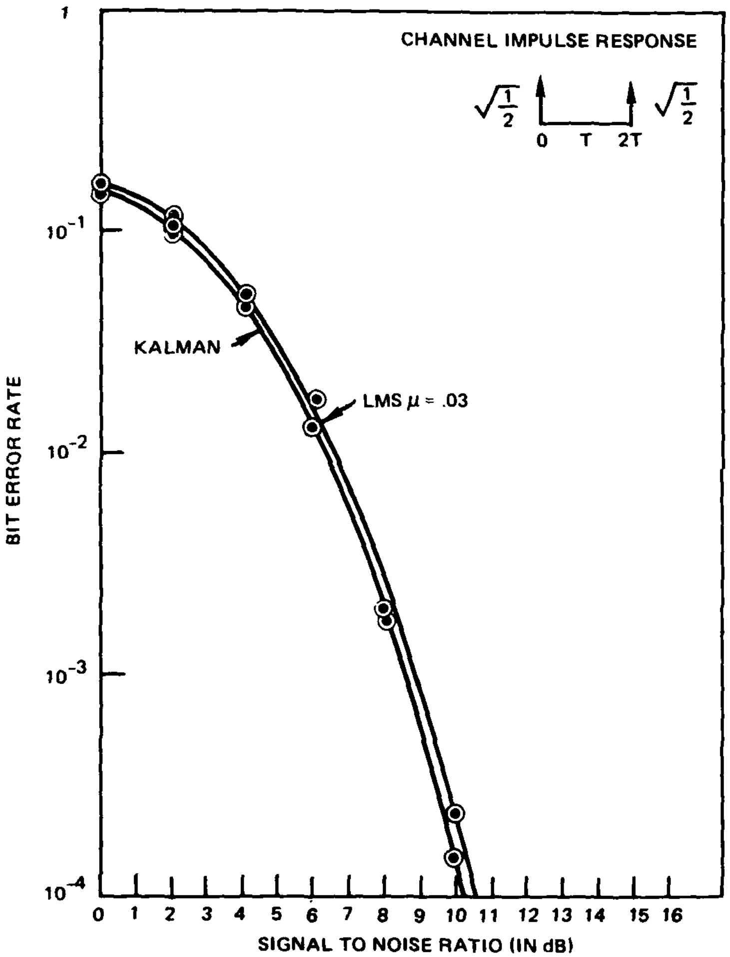
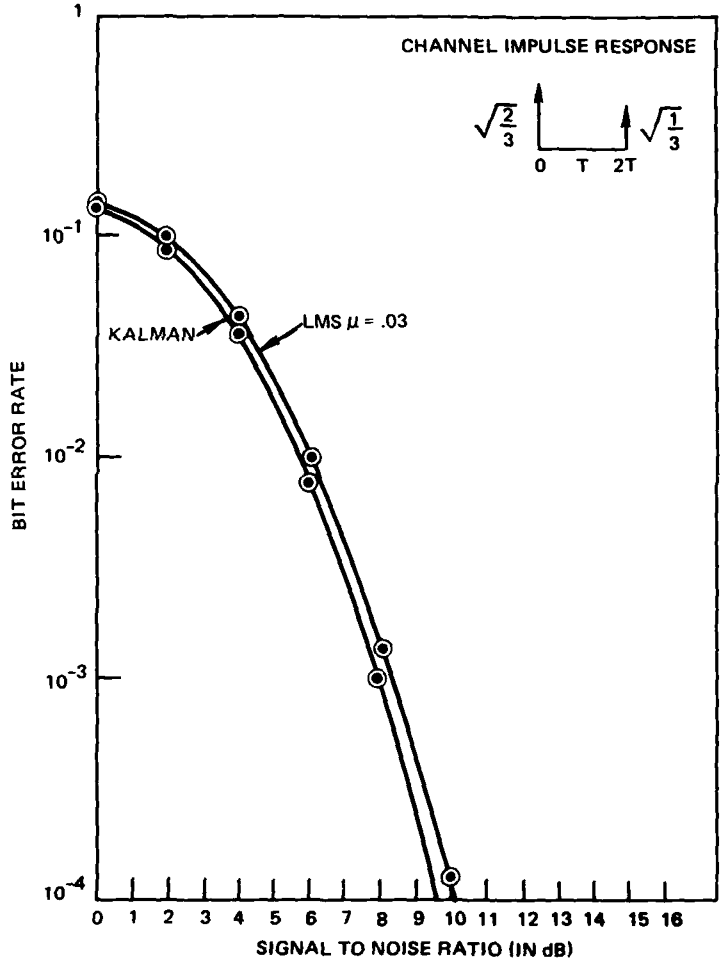


Figure 18. MLSE performance for preamble of length 1000 for two cases of channel impulse response and using LMS and Kalman filter channel estimators.



a. MLSE performance for preamble length 50.



b. MLSE performance for preamble length 50.

Figure 19. MLSE performance for preamble of length 50 for two cases of channel impulse response and using LMS and Kalman filter channel estimators.

shorter preamble length compensates for the less than 0.5 dB loss in performance. The LMS filter, using the best μ value, is not significantly degraded either. Figure 19 also shows that the degradation caused by using the shorter preamble lengths with the LMS filter is no more than 0.5 dB. Still the Kalman filter performs better than the LMS filter. This is to be expected since the Kalman filter was designed to minimize the mean square error at each iteration of the algorithm while the LMS filter uses a noisy gradient technique. Figure 19 demonstrates that a shorter preamble length is a definite possibility for use with the MLSE. Depending on the situation in which one uses the MLSE one would choose the proper estimator algorithm and preamble length to achieve the desired goals under the operational constraints of the system. This choice would take several things under consideration. Included should be the amount of computation allowable, the degree of accuracy desired, and the length of time to be allowed for the estimation process.

In some situations accuracy may not be so important a design criterion as computational burden when selecting an estimator algorithm. In this type of situation one might choose the LMS estimation algorithm. The important thing to remember when using the LMS is that the proper value for μ must be used. Figure 20 shows what can happen when a poor value of μ is used. Here, the optimal value and a suboptimal value are used with a 50-bit preamble and a 1000-bit preamble. As can be seen, the results are poor at best, and for preamble length 50 the results are disastrous. This demonstrates the importance of exercising discretion in choosing a value for μ.

In this section the problem of the unknown channel was solved for the Maximum Likelihood Sequence Estimator. Performance curves for the MLSE using a channel estimator are given. The use of a channel estimator does not seriously degrade the expected performance of the MLSE, as can be seen in figure 21.* Judicious selection of the estimator algorithm, preamble length, and values for μ (for the LMS) must be made in order to ensure good performance of the MLSE under operational conditions. The longer the preamble, the better the performance. The Kalman filter is the best estimator to use. However, it suffers from the disadvantage that the number of computations at each iteration of the algorithm is proportional to the square of the number of parameters to be estimated. The LMS eases the computational burden, the number of computations required at each iteration being linearly proportional to the number of parameters. However, it is not optimal and the parameter μ must be chosen carefully.

V. MLSE FOR TIME VARYING CHANNELS

The development and analysis of the Maximum Likelihood Sequence Estimator up to this point has assumed that the channel is time invariant. In many real-time scenarios this is not a valid assumption. However, the channel is often changing slowly enough so that the variations in time will have little effect on the performance of the MLSE. Alternatively, the message sequence may be short enough so that the variations in time do not affect it. The work done on the MLSE up to now has been for those channels in which variations in time would have little or no effect on the performance of the MLSE. This section extends the MLSE to the case in which time variations occur, the adaptive MLSE. All the examples in this section use the Kalman filter to estimate the channel response during the preamble and also during the periodic updates.

*Here, the preamble consists of 100 bits and the message of a million bits.

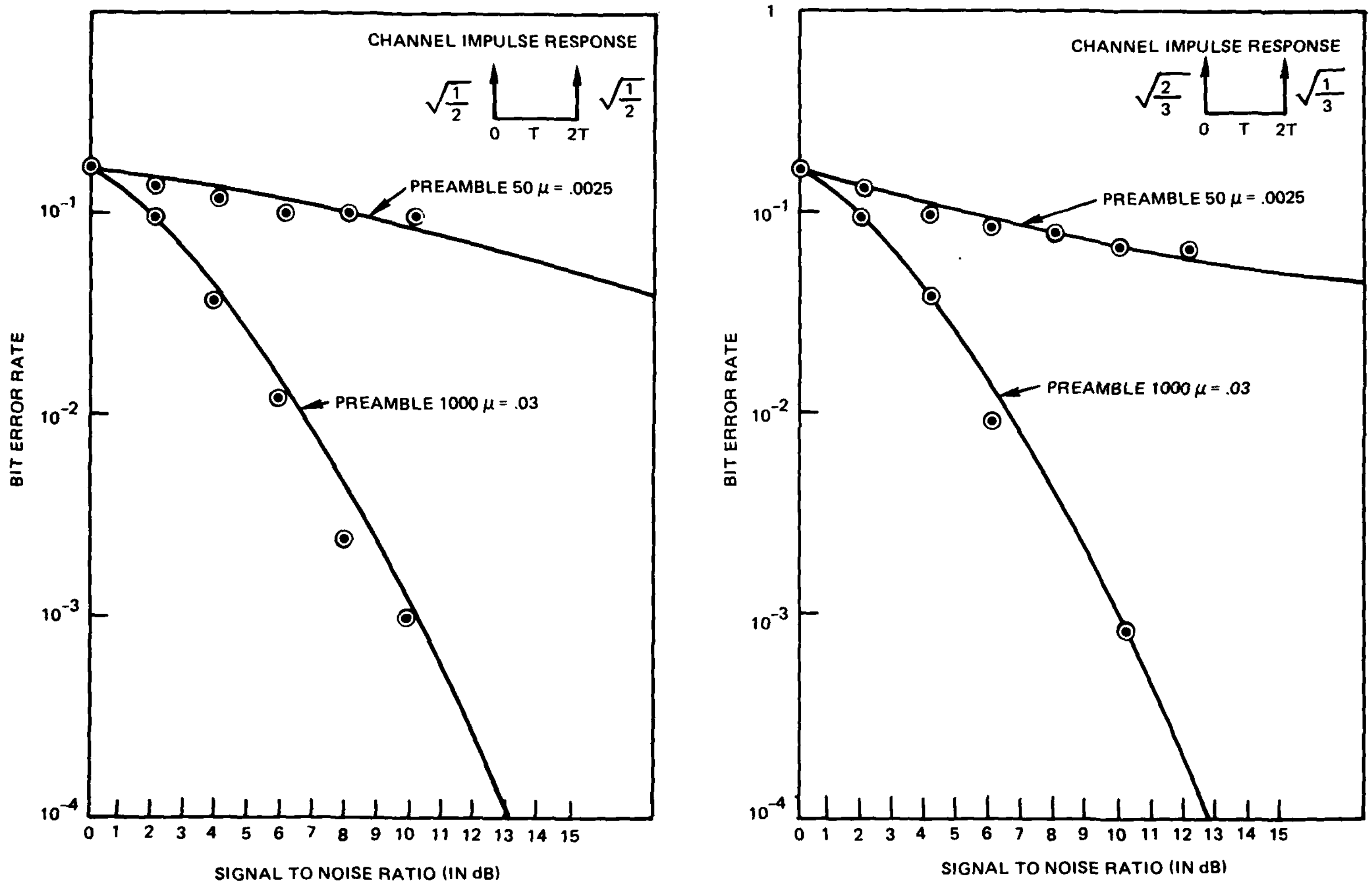


Figure 20. MLSE performance comparison using optimal and suboptimal values of μ for an LMS filter.

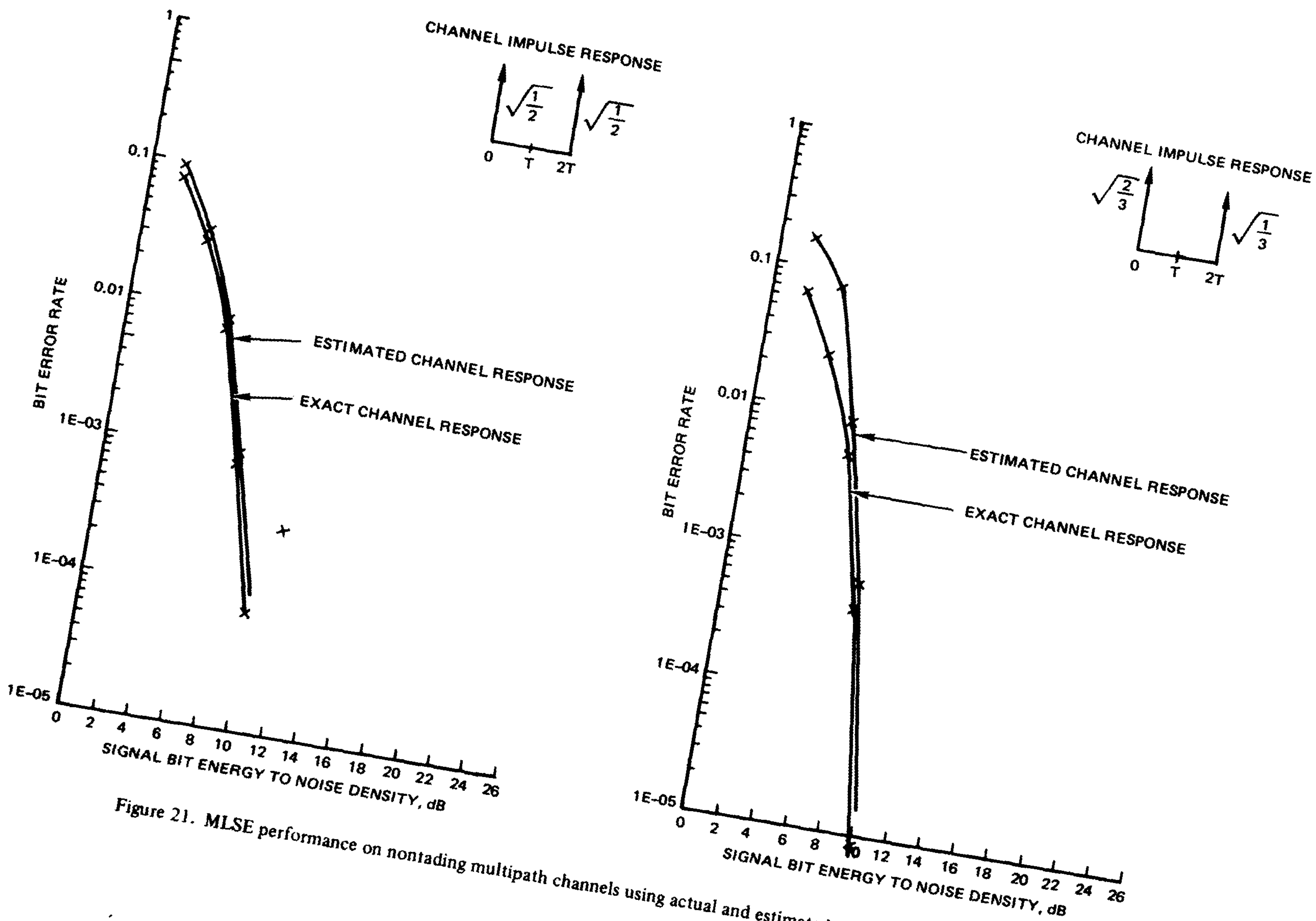


Figure 21. MLSE performance on nonfading multipath channels using actual and estimated channel response.

Time variations of the channel (response) will often degrade the performance of the MLSE to the point at which communication across the channel is extremely poor if not impossible. The variations in channel response will cause the matched filter, which up to this point has been fixed in time, to become mismatched. When this happens the output of the matched filter is no longer the correct input to the Viterbi decoder. As can be seen in reference (17), this causes degradation in the performance of the MLSE. The matched filter must be modified or updated to keep track of changes in channel response. Alternative approaches to this problem have been discussed by others in references (18–20).

The approach used here will be referred to as periodic tracking. Here the data stream must be interrupted and a known pseudonoise (PN) sequence must be transmitted. This PN sequence has to be known at the receiver as well as at the transmitter. During this burst of known data the receiver will re-estimate the channel response, set up the matched filter, and re-initialize the metrics. This could be a problem in some cases. It may be undesirable to interrupt the information sequence and even more undesirable to send a known sequence interleaved with the information sequence. Another problem will be that of determining at what intervals the known sequence should be sent. This would probably be a function of signal-to-noise ratio, rate of change of the channel, and length of the total message.

The results obtained for the performance of the MLSE in sections III and IV were obtained by simulating the signals and noise on a computer. The results in this section were obtained by using data from an on-the-air-hf field test. The field test was conducted by transmitting signals over the hf channel from Point Mugu to Point Loma. The tests were conducted at different times of day in order to get varying channel responses. With different channel responses the MLSE algorithm could be tested for different multipath structures. It was also done this way to test the algorithm on enough different channels to see whether this approach would be feasible for the Navy to use as a communications system. A known pseudorandom number sequence was used to generate the sequence of bits sent over the channel. The sequence was sent at a high signal-to-noise ratio. At the receiver the signals were recorded onto tape along with information concerning which sequence was sent. The tapes were processed at a later time on the Univac to obtain the curve for signal-to-noise ratio versus bit error rate obtained in the earlier sections. This was done by assuming that the signal tapes were at a high enough signal-to-noise ratio that they could be considered as pure signal—a simplification which made it possible to obtain the bit error rates at varying signal-to-noise ratios. The signals plus noise were obtained by adding white Gaussian noise to the data from the signal tapes with appropriate weightings to obtain the desired signal-to-noise ratios. As can be seen in figure 24, the recorded SNR limited the simulation curves in some cases. Our main interest was to see whether these curves were comparable to the curves we obtained via simulation. For each of the recorded signals in figures 22–25, appendix A contains additional information on the channel pulse response (matched filter output) and signal-to-noise ratio (ref 21).

---

[18] Ungerboeck, G, Adaptive Maximum-Likelihood Receiver for Carrier-Modulated Data Transmission Systems, <u>IEEE Transactions on Communication</u>, vol COM-22, no 5, MAy 1974

[19] Magee, FR Jr, A Comparison of Compromise Viterbi Algorithms and Standard Equalization Technique Techniques over Band-Limited Channels, <u>IEEE Transactions on Communication</u>, vol COM-23, no 3, March 1975

[20] Falconer, DD, & FR Magee, Jr, Adaptive Channel Memory Truncation for Maximum-Likelihood Sequence Estimation, <u>Bell System Technical Journal</u>, vol 9, p 1541-1562, November 1973

[21] Hoff, LE, AR King, <u>Skywave Communication Techniques</u>, NOSC Technical Report 709, September 1981

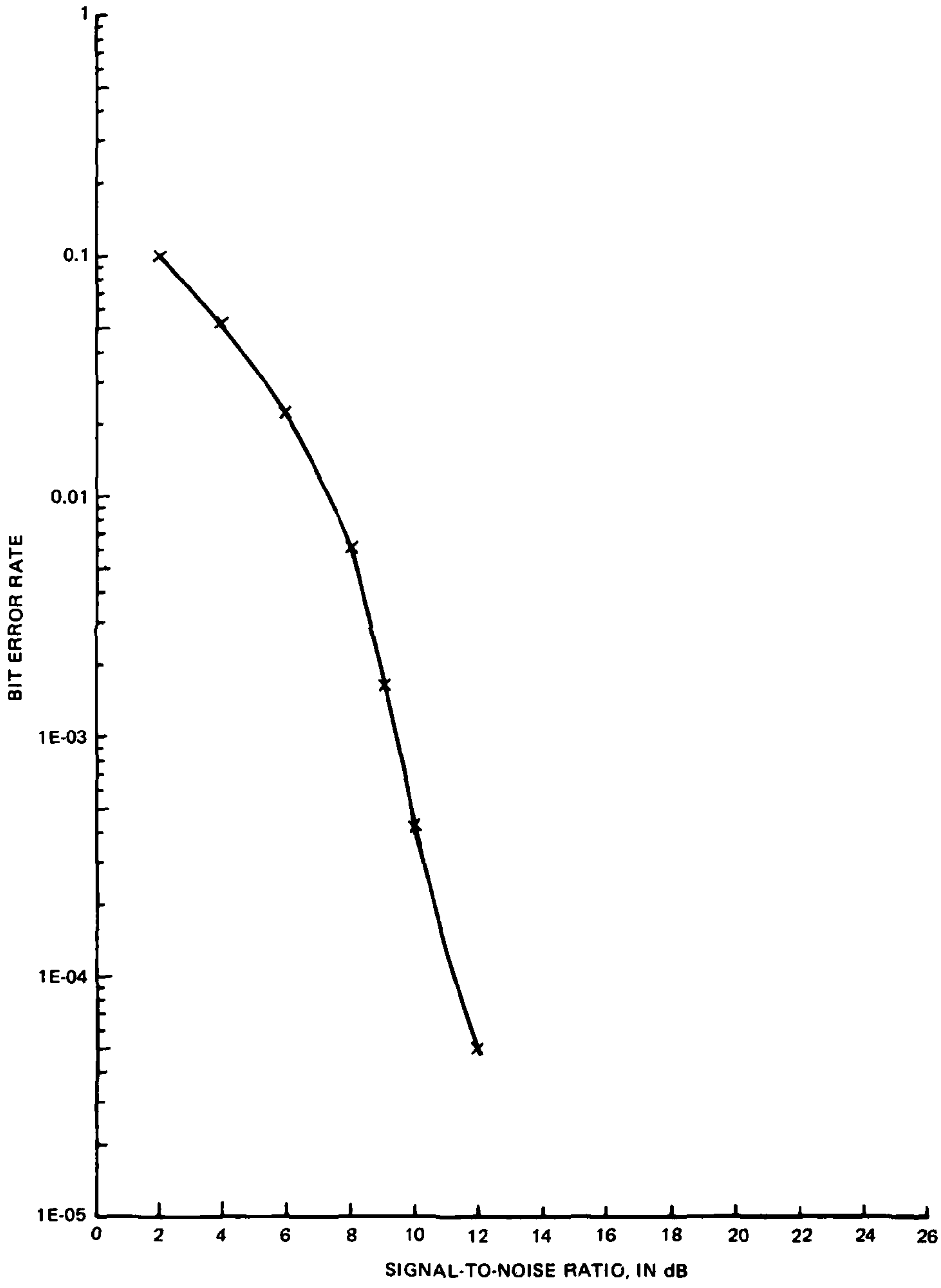


Figure 22. Performance of nonadaptive MLSE on hf data obtained 13 May at 1030.

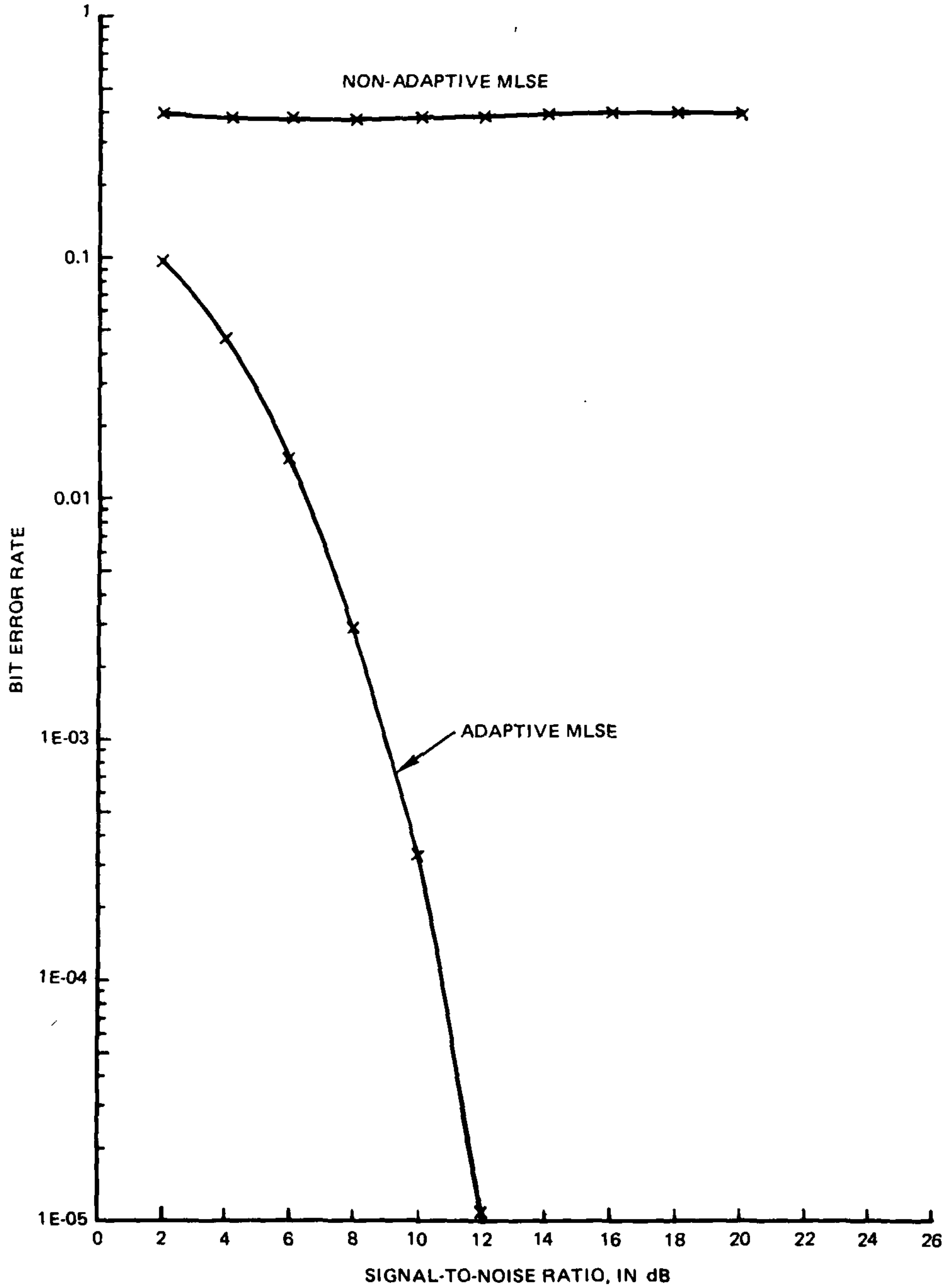


Figure 23. Performance of adaptive and nonadaptive MLSE on hf data obtained 13 May at 1740.

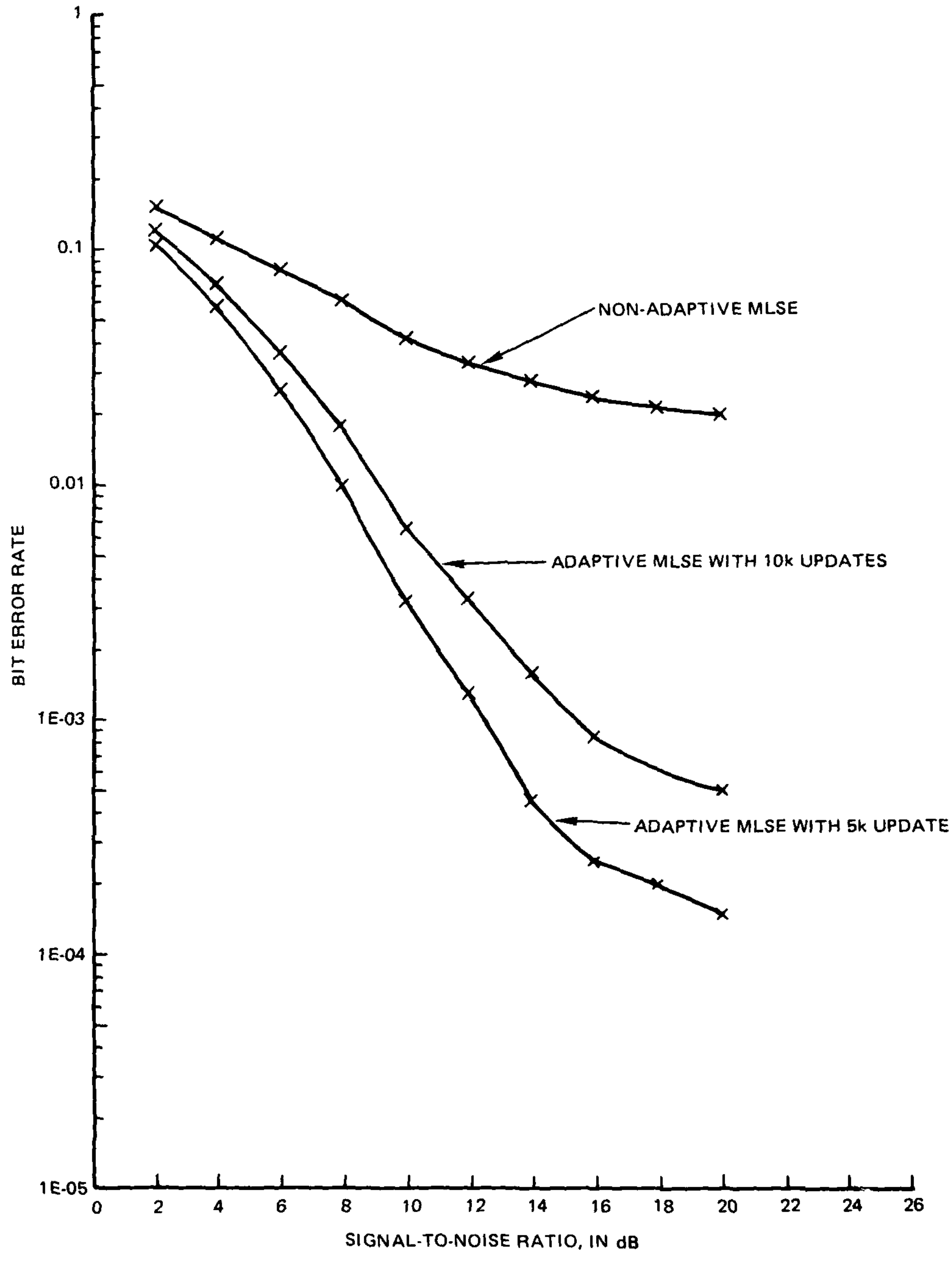


Figure 24. Performance of adaptive and nonadaptive MLSE on hf data obtained 14 May at 1720. Recorded signal-to-noise ratio was 17 dB, which is therefore the maximum SNR available for simulation.

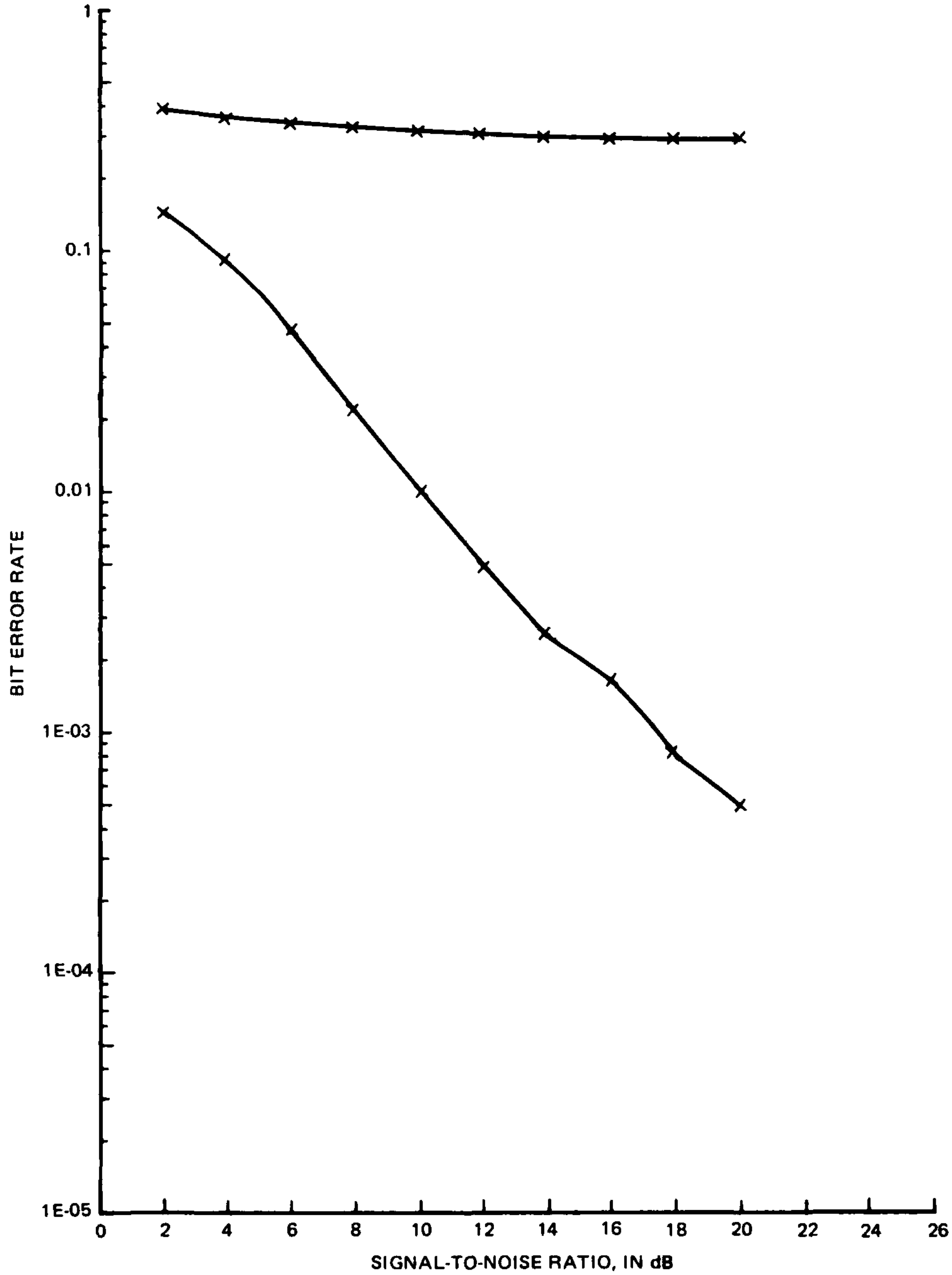


Figure 25. Performance of adaptive and nonadaptive MLSE on hf data obtained 13 May at 1620.

The first example of MLSE performance on real data is given in figure 22. This example of an hf channel is a simple one; the channel consists of only the surface wave. It does not exhibit any multipath structure nor does it exhibit any time variations. Figure 22 shows the performance of the MLSE on this simple channel for a signal lasting 41 seconds or 99000 bits. The nonadaptive MLSE performs well for this case in which no multipath or fading is present.

The next example of MLSE performance on real data is for a channel exhibiting both a surface wave and a skywave. Here the two paths are separated by .3 ms. The first path (surface wave) is down 10 dB in power from the second path. Figure 23 shows the performance results for the nonadaptive MLSE as well as for the adaptive MLSE. The results were obtained for a signal lasting 37 seconds. The performance of the adaptive MLSE is clearly better than the performance of the nonadaptive MLSE. This indicates that the channel exhibits some sort of time variation over the 37 seconds. The adaptive MLSE is able to decode the output of this unknown, time variant channel with good results. Otherwise, even when using the nonadaptive MLSE, reliable communication over this channel would be impossible.

Figure 24 shows the performance of the MLSE on a third hf channel for a signal lasting 25 seconds. Here, the nonadaptive MLSE is compared with the adaptive MLSE with two different update periods. The shorter the update period (or the faster the update rate), the better the performance. This channel consists of a single path which is not fading but is exhibiting some phase variations.

The fourth example is a channel consisting of a surface wave and two skywaves. Here the surface wave is down 5 dB from the strongest skywave, which is delayed by .3 ms. The second skywave is delayed by 2 ms and is 13 dB down from the first skywave. This channel exhibits fading. Figure 25 shows the performance of the nonadaptive MLSE and the adaptive MLSE with 5k updates. Here, the improvement obtained using the adaptive MLSE is significant.

Results obtained from an hf field test using the MLSE to decode the received signal show that this approach to processing the data received over unknown, time varying multipath channels gives good results.

The fifth example is a channel consisting of a surface wave and a skywave. Here, the surface wave is 10 dB down in power from the skywave.* Some fading occurs in this example but the adaptive MLSE handles the fading well, as can be seen in figure 26. Figure 26 shows the performance of the nonadaptive MLSE versus the performance of the adaptive MLSE with two update rates. The MLSE with update rate of 5k bits does slightly better than the MLSE with update rate of 10k bits.

VI. CONCLUSIONS AND RECOMMENDATIONS

CONCLUSIONS

The MLSE can be made adaptive by interleaving short reference sequences with the message sequence. An estimation algorithm is applied to these sequences to estimate the changing channel response. Tests on simulated nonfading multipath channels demonstrate that the adaptive MLSE is an effective algorithm for unknown, nonfading multipath channels

*The skywave is delayed .3 ms from the surface wave.

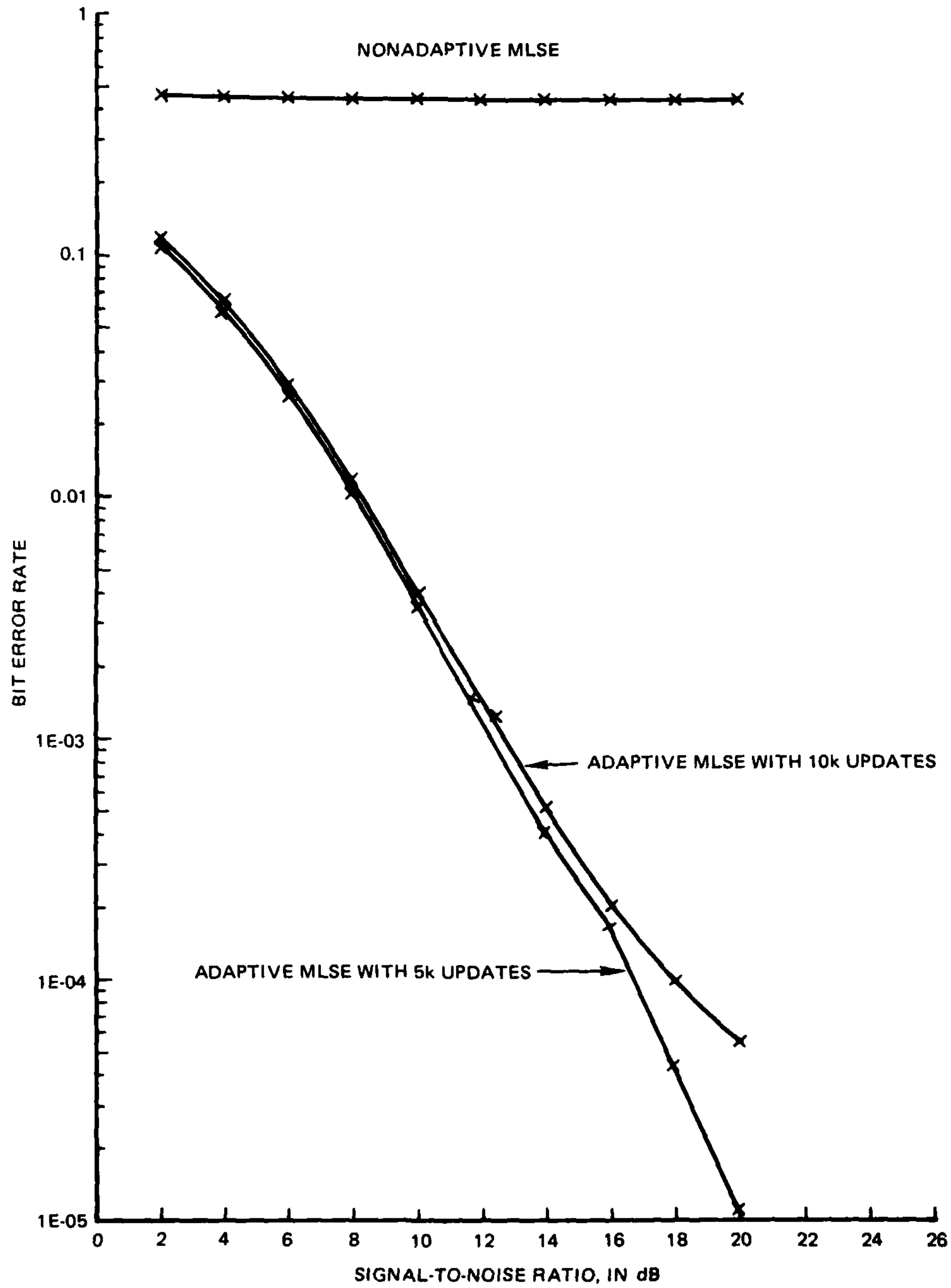


Figure 26. Performance of adaptive and nonadaptive MLSE on hf data obtained 13 May 1981 at 1210.

and that it is superior to signaling schemes currently used for multipath channels. By using signals recorded during an hf field test, we have shown that the adaptive MLSE can be extended to time variant (fading) multipath channels. The error rate performance depends upon the channel rate of change and the channel estimate update rate.

RECOMMENDATIONS

1. Further testing of the adaptive MLSE on time variant multipath channels is necessary to define channel estimate update rates. A careful comparison of the adaptive MLSE and other signaling schemes would provide valuable information to system engineers.

2. Because of the very fast rate of change of some military channels, a continuous tracking technique should be developed for the adaptive MLSE.

3. Because of the large dispersion found on some military communication channels, the MLSE using the Viterbi algorithm will not be practical in many cases. Sequential decoding algorithms, whose complexity grows only linearly with channel dispersion, should be developed for the MLSE algorithm (ref 22).

4. The MLSE should be tested on channels with impulsive noise and the bit error rate results analyzed. The sensitivity of the MLSE algorithm to bursts of impulse noise should be determined.

[22] Hoff, LE and DR Bean, The Stack Algorithm for Sequential Decoder, NOSC Technical Note 924, 30 October 1980

REFERENCES

1. G David Fornery, Jr, Maximum-Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference, IEEE Transactions on Information Theory, vol IT-18, no 3, May 1972

2. G Ungerboeck, Adaptive Maximum Likelihood Receiver for Carrier Modulated Data-transmission Systems, IEEE Transactions on Communications, vol COMM-22, no 5, May 1974

3. G Ungerboeck, Linear Receiver and Maximum-Likelihood Sequence Receiver for Synchronous Data Signals, IEEE International Communications Conference Proceedings, June 1973

4. Morley, RE, Jr, DL Snyder, Maximum Likelihood Sequence Estimation for Randomly Dispersive Channels, IEEE Transactions on Communications, vol COM-27, no 6, June 1979

5. Hoff, LE, RL Merk, Soft Decision Demodulation Using the Viterbi Algorithm, NOSC Technical Note 544,* September 1978

6. Norvell, S Channel Estimation for the HF Channel, NOSC Technical Note 545,* September 1978

7. Hoff, LE, Norvell, S, Adaptive Maximum Likelihood Sequence Estimation for the HF Channel, 13th Asilomar Conference on Circuits, Systems and Computers, November 1979

8. Andrew J Viterbi, Convolutional Codes and their Performance in Communications Systems, IEEE Transactions on Communications Technology, vol COM-19, no 5, October 1971

9. Carl W Helstrom, Statistical Theory of Signal Detection, Pergamon Press. Second Revised edition, Hungary, 1968

10. ML Doelz, ET Heald and DL Martin, Binary Data Transmission Techniques for Linear Systems, Proceedings IRE, vol 45, p 656–661, May 1957

11. CR Cahn, Combined Digital Amplitude and Phase Modulation, IRE Transactions on Communications Systems, vol CS-8, p 150–155, September 1960

12. Pennoyer, BL, Performance of an Adaptive or Mismatched Maximum Likelihood Sequence Estimation Receiver, PhD thesis, University of Southern California, February 1977

13. HW Sorenson, Kalman Filtering Techniques, Advances in Control Systems, vol 3, p 219–292 1966

14. HW Sorenson, Filtering and Random Processes in Control, class notes

*NOSC technical notes are informal documents intended chiefly for internal use.

15. B Widrow, et al, Stationary and Nonstationary Learning Characteristics of the LMS Adaptive Filter, Proceedings of the IEEE, vol 64, no 8, 8 August 1976, p1151-1162

16. B Widrow, Adaptive Filters I: Fundamentals, SEL-66-126 Stanford Electronics Laboratories, Stanford, California, December 1966

17. Magee, FF Jr, JG Proakis, Adaptive Maximum-Likelihood Sequence Estimation for Digital Signaling in the Presence of Intersymbol Interference, IEEE Transactions on Information Theory, vol IT-19, January 1973

18. Ungerboeck, G, Adaptive Maximum-Likelihood Receiver for Carrier-Modulated Data Transmission Systems, IEEE Transactions on Communication, vol COM-22, no 5, May 1974

19. Magee, FR, Jr, A Comparison of Compromise Viterbi Algorithms and Standard Equalization Techniques over Band-Limited Channels, IEEE Transactions on Communication, vol COM-23, no 3, March 1975

20. Falconer, DD, and FR Magee, Jr, Adaptive Channel Memory Truncation for Maximum-Likelihood Sequence Estimation, Bell System Technical Journal, vol 9, p 1541-1562, November 1973

21. Hoff, LE, AR King, Skywave Communication Techniques, NOSC Technical Report 709, September 1981

22. Hoff, LE and DR Bean, The Stack Algorithm for Sequential Decoder, NOSC Technical Note 924,* 30 October 1980

APPENDIX A: STATISTICS OF RECORDED HF SIGNALS

Table A-1 summarizes the characteristics of the recorded signals in figures A-1 through A-5.

Day-Time	Mode	Delay, ms	Power, dB	Freq, MHz	RF BW, kHz	Tape (S+N)/, dB
May 13 1030	S	0	0	3.357	2.4	22.1
	S	0	-10		2.4	24.4
May 13 1210	E	0.3	0	6.835		
	S	0	- 5		2.4	17.1
May 13 1620	E	0.3	0	5.785		
	F	2.0	-13			
	S	0	-10		2.4	27.3
May 13 1740	E	0.3	0	5.785		
May 14 1720	S	0	0		2.4	20.6

Table A-1. Characteristics of recorded hf signals.

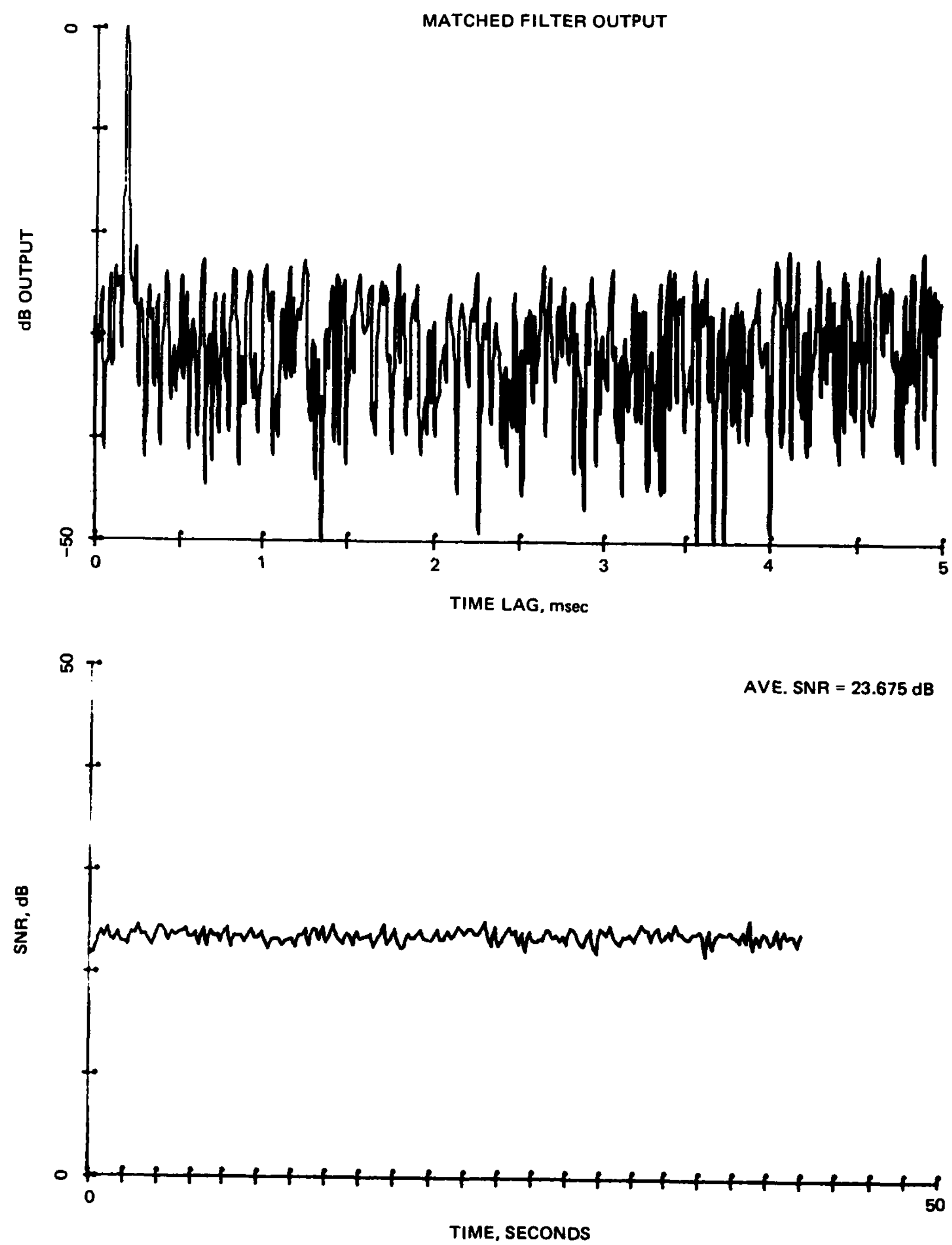


Figure A-1. Hf signal recorded 13 May 1980 at 1030.

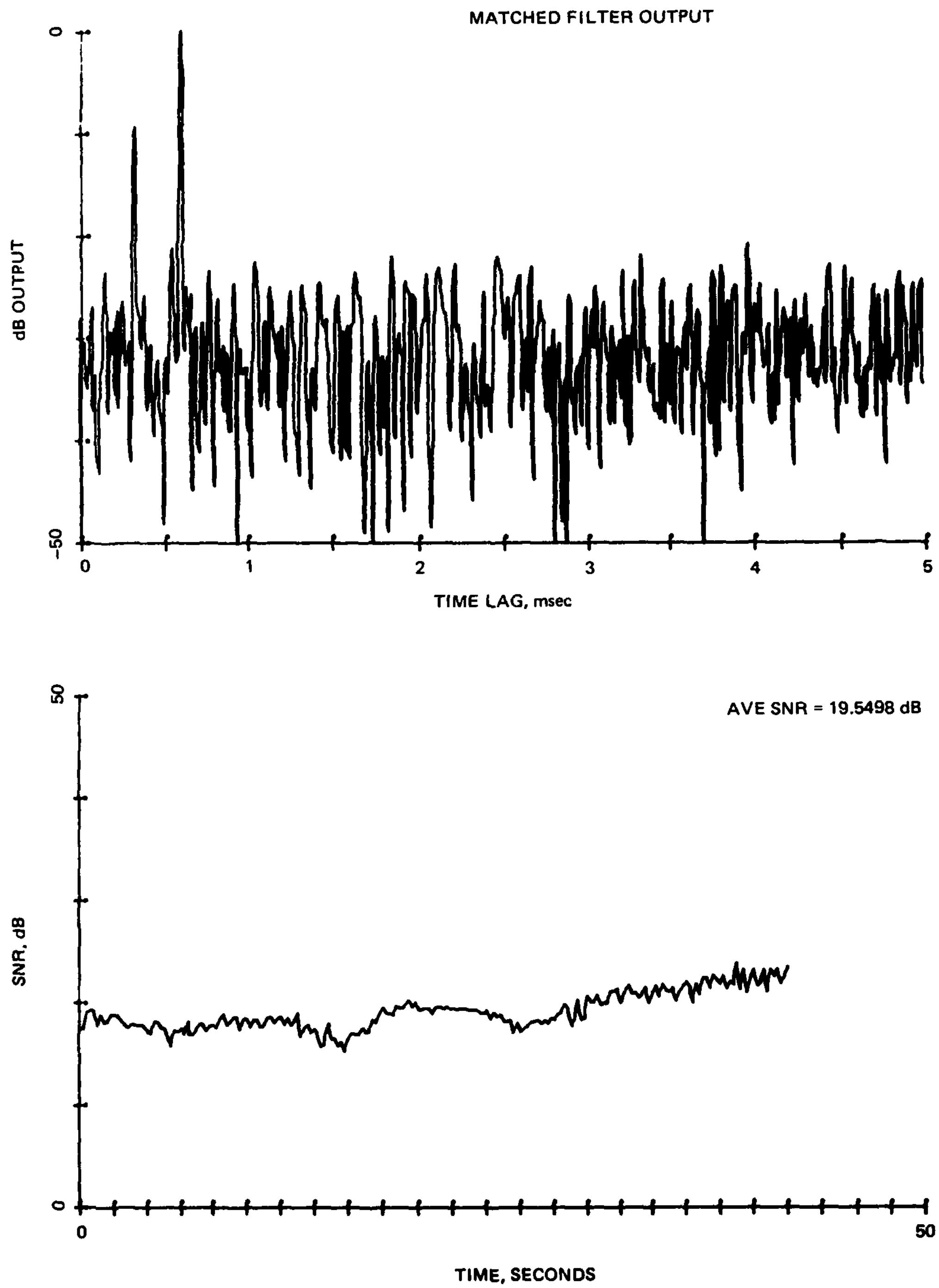
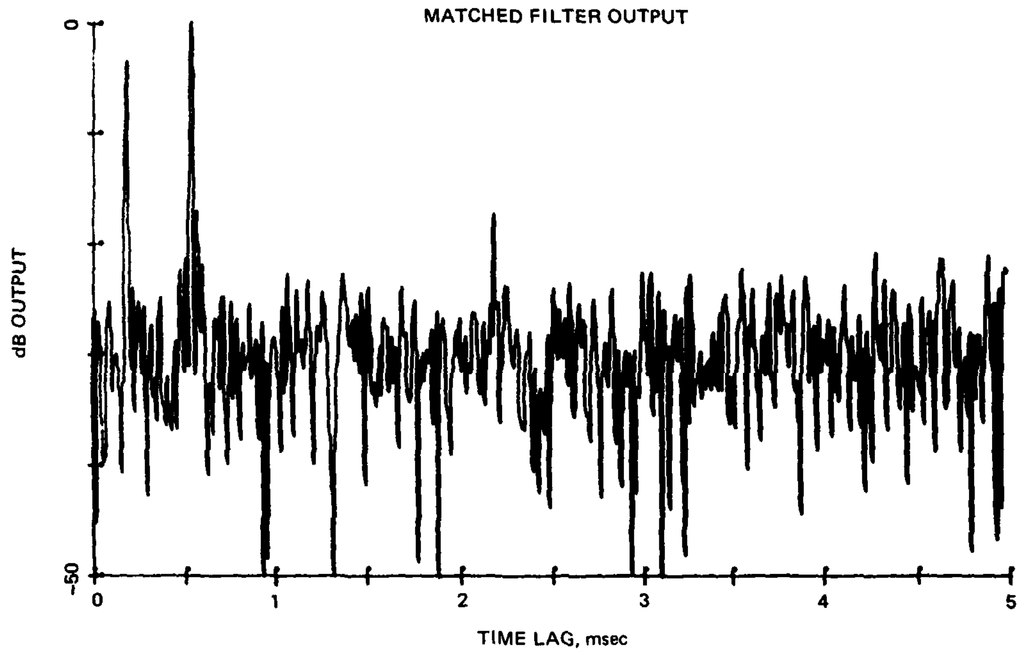


Figure A-2. Hf signal recorded 13 May 1980 at 1210.



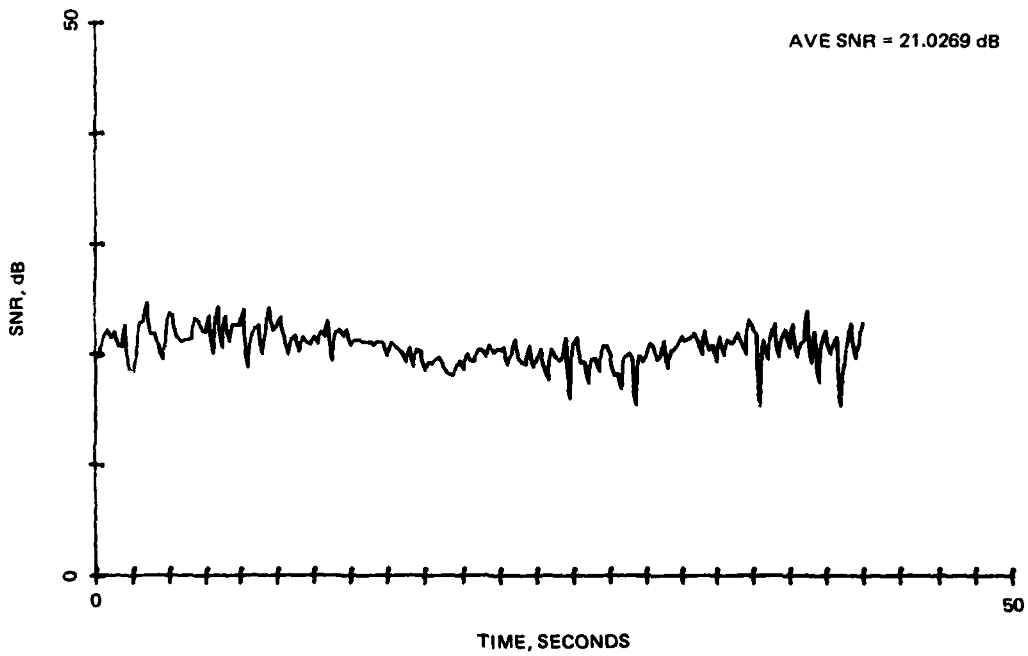


Figure A-3. Hf signal recorded 13 May 1980 at 1620.

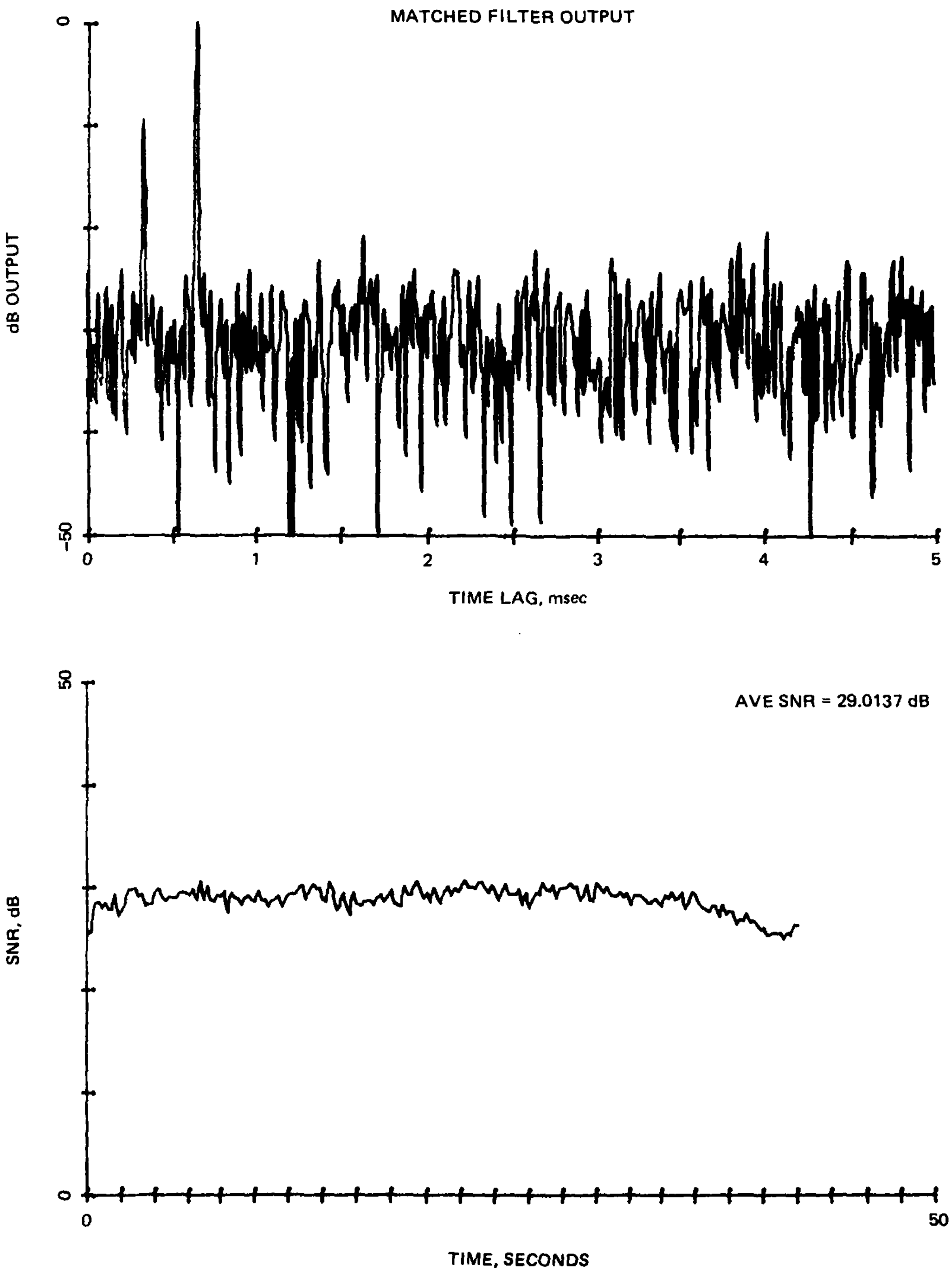
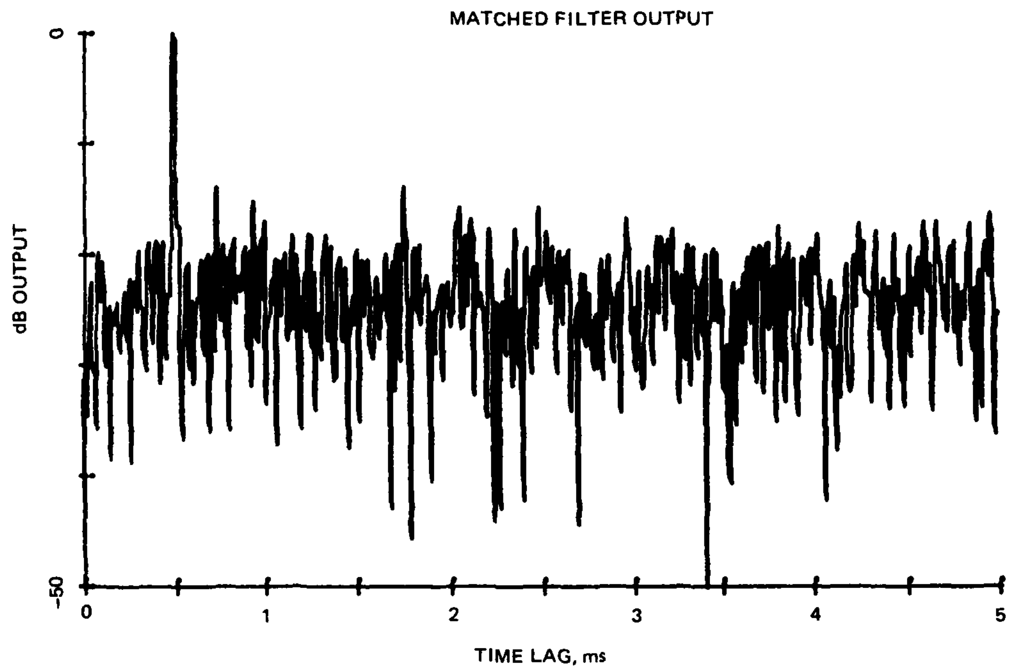


Figure A-4. Hf signal recorded 13 May 1980 at 1740.



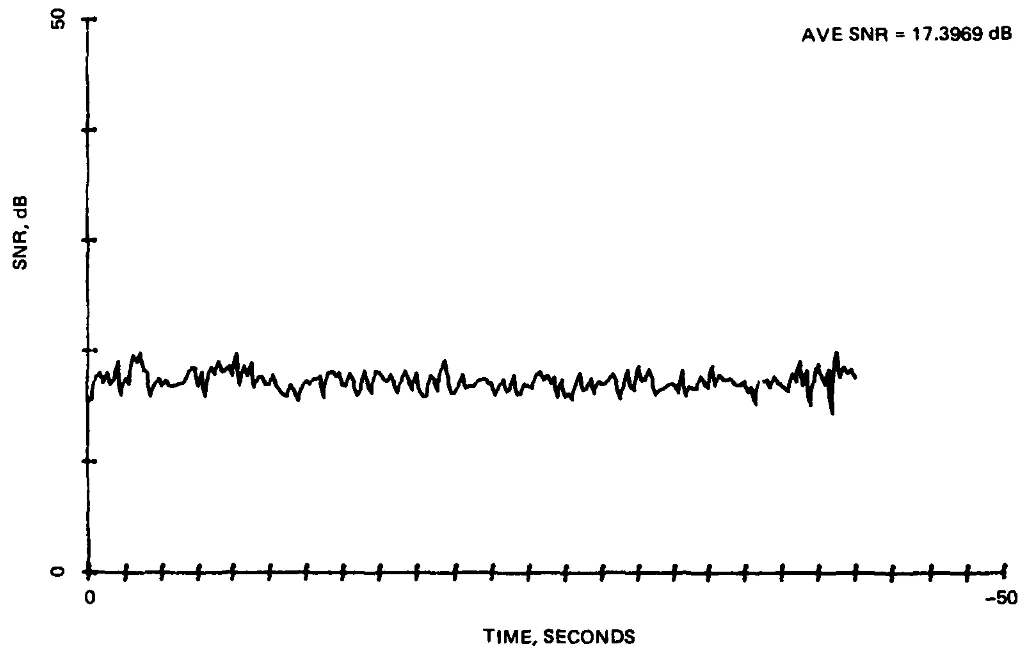


Figure A-5. Hf signal recorded 14 May 1980 at 1720.

ATE

LME